01-05-10

DAC / IFW

12/22/2009

USPTO

Office of Petitions

P.O. Box 1450

Alexandria, VA.

Subject: Set Aside Abandonment of Application # 09/945, 176

Mr. Tod Swann

Enclosed are revised Claims and supporting documents that show my patent application #09/945,176 has been grievous mishandled!

Patent application #09/945,176 Docketed to Examiner in the GAU on 01/19/2002, 03/08/2002, 08/13/2003, 09/17/2004. Again, in 11/23/2004 docketed to another examiner. An Office Action Summary was prepared by Mr. David Jung and on 11/30/2004 was mailed to me for review as a Non-Final Rejection. The focus of the rejection was the Claims section of the application. All Seven of the Claims in my pending patent were rejected.

Upon receipt of these documents, I proceeded to review the disputed section and decided on a complete change to the Claims section. An amendment to the Claims section of the application was prepared indicating my changes. These and supporting documents were sent by Certified Mail to Mr. Jung on 12/16/2004. I received a certificate of receipt on 12/20/2004.

I received no further communication from Mr. Jung. The transaction history remained the same through 2004 and into 2005. Concerned with the disposition of my patent application I called the Patent Assistance Center. I was informed that the time had expired for response to the Non-Final Rejection and that my patent might be abandoned. Alarmed, I tried to talk to Mr. Jung but was unable to get any response. Finally I called Mr. Greg Morse and he said he would look into the matter and let me know by mail. I received no communication from either Mr. Jung or Mr. Morse.

On 02/23/2006 I called Mr. Morse and he said he would have Mr. Jung call me. The same day, Mr. Jung called but had no information about the application. In point of fact he was almost totally incoherent. His manner on the phone gave the impression that he did not know anything about my patent application? The only value I received from his call was a phone number [1-800-786-9199].

I am 82 years old and living on social security and SSI. During the period described in this letter, problems occurred that resulted in the failure of the general health of my heart and the onset of a permanent physical disability. Between the period of 11/30/2004 and 12/30/2006 I underwent several major medical procedures and was in the process of recuperation when on 02/04/08 I received the final notice of abandonment of my patent application from David Y. Young. From past actions (see above) this represented an apparent culmination of continued attempts to undermine and eventually abandon my patent application.

This letter is a formal request that the current status of Abandonment for patent application # 09/945,176 be considered as inappropriate. That it be considered as timely and that the process for its acceptance be continued.

Respectfully,

Donald S. Merrill

D S Merrill

SEQUENCE LISTING

DECLARATION — Utility or Design Patent Application

I hereby claim the benefit under 35 U.S.C 120 of any United States application(s), or 365(c) of any PCT international application designating the United States of America, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT International application in the manner provided by the first paragraph of 35 U.S.C 112, I acknowledge the duty to disclose information which is material to patentability as defined in 37 CFR 1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application

U.S. Parent Application or PCT Parent Number	Parent Filing Date (MM/DD/YYYY)	Parent Patent Number *(if applicable)*

☐ Additional U S or PCT international application numbers are listed on a supplemental priority data sheet PTO/SB/02B attached hereto

As a named inventor, I hereby appoint the following registered practitioner(s) to prosecute this application and to transact all business in the Patent and Trademark Office connected therewith ☐ Customer Number [ ] → *Place Customer Number Bar Code Label here*
OR
☐ Registered practitioner(s) name/registration number listed below

Name	Registration Number	Name	Registration Number

☐ Additional registered practitioner(s) named on supplemental Registered Practitioner Information sheet PTO/SB/02C attached hereto

Direct all correspondence to ☐ Customer Number or Bar Code Label [ ] *OR* [+] Correspondence address below

Name	Donald Sidney Merrill				
Address	294 Rio Lindo Ave				
Address	Apartment 41				
City	Chico	State	CA	ZIP	95926-5507
Country	United States	Telephone	(530) 897-0735	Fax	

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C 1001 and that such willful false statements may jeopardize the validity of the application or any patent issued thereon

Name of Sole or First Inventor: ☐ A petition has been filed for this unsigned inventor

Given Name (first and middle [if any])				Family Name or Surname			
Donald Sidney Merrill							
Inventor's Signature	D S Merrill					Date	8/29/01
Residence: City	Chico	State	CA.	Country	United States	Citizenship	U.S.
Post Office Address	294 Rio Lindo Ave.						
Post Office Address	Apartment 41						
City	Chico	State	CA	ZIP	95926-5507	Country	U. S.

☐ Additional inventors are being named on the ____ supplemental Additional Inventor(s) sheet(s) PTO/SB/02A attached hereto

JAN 04 2010

Please type a plus sign (+) inside this box → [+]

PTO/SB/05 (4/98)
Approved for use through 09/30/2000 OMB 0651-0032
Patent and Trademark Office. U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 C.F.R. § 1.53(b))

Attorney Docket No.	
First Inventor or Application Identifier	Donald S. Merrill
Title	Electronic Facilitation Venue
Express Mail Label No	

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

ADDRESS TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231

1. [+] * Fee Transmittal Form (e.g., PTO/SB/17) (Submit an original and a duplicate for fee processing)
2. [+] Specification [Total Pages 3] (preferred arrangement set forth below)
   - Descriptive title of the Invention
   - Cross References to Related Applications
   - Statement Regarding Fed sponsored R & D
   - Reference to Microfiche Appendix
   - Background of the Invention
   - Brief Summary of the Invention
   - Brief Description of the Drawings (if filed)
   - Detailed Description
   - Claim(s)
   - Abstract of the Disclosure
3. [+] Drawing(s) (35 U.S.C. 113) [Total Sheets 10]
4. Oath or Declaration [Total Pages 2]
   - a. [+] Newly executed (original or copy)
   - b. [ ] Copy from a prior application (37 C.F.R. § 1.63(d)) (for continuation/divisional with Box 16 completed)
     - i. [ ] DELETION OF INVENTOR(S) Signed statement attached deleting inventor(s) named in the prior application, see 37 C.F.R. §§ 1.63(d)(2) and 1.33(b).

* NOTE FOR ITEMS 1 & 13: IN ORDER TO BE ENTITLED TO PAY SMALL ENTITY FEES, A SMALL ENTITY STATEMENT IS REQUIRED (37 C.F.R. § 1.27), EXCEPT IF ONE FILED IN A PRIOR APPLICATION IS RELIED UPON (37 C.F.R. § 1.28).

5. [ ] Microfiche Computer Program (Appendix)
6. Nucleotide and/or Amino Acid Sequence Submission (if applicable, all necessary)
   - a. [ ] Computer Readable Copy
   - b. [ ] Paper Copy (identical to computer copy)
   - c. [ ] Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

7. [ ] Assignment Papers (cover sheet & document(s))
8. [ ] 37 C.F.R. § 3.73(b) Statement (when there is an assignee) [ ] Power of Attorney
9. [ ] English Translation Document (if applicable)
10. [ ] Information Disclosure Statement (IDS)/PTO-1449 [ ] Copies of IDS Citations
11. [ ] Preliminary Amendment
12. [ ] Return Receipt Postcard (MPEP 503) (Should be specifically itemized)
13. [ ] * Small Entity Statement(s) (PTO/SB/09-12) [ ] Statement filed in prior application, Status still proper and desired
14. [ ] Certified Copy of Priority Document(s) (if foreign priority is claimed)
15. [ ] Other: ....

16. If a CONTINUING APPLICATION, check appropriate box, and supply the requisite information below and in a preliminary amendment

[ ] Continuation [ ] Divisional [ ] Continuation-in-part (CIP) of prior application No: ______/______

Prior application information: Examiner ______ Group / Art Unit ______

For CONTINUATION or DIVISIONAL APPS only: The entire disclosure of the prior application, from which an oath or declaration is supplied under Box 4b, is considered a part of the disclosure of the accompanying continuation or divisional application and is hereby incorporated by reference. The incorporation can only be relied upon when a portion has been inadvertently omitted from the submitted application parts.

17. CORRESPONDENCE ADDRESS

[ ] Customer Number or Bar Code Label (Insert Customer No. or Attach bar code label here) or [+] Correspondence address below

Name	Donald Sidney Merrill				
Address	294 Rio Lindo Ave.				
	Apartment 41				
City	Chico	State	California	Zip Code	95926-5507
Country	United States	Telephone	(530) 897-5507	Fax	

Name (Print/Type)	Donald Sidney Merrill	Registration No (Attorney/Agent)	
Signature		Date	

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
WASHINGTON, D.C. 20231
www.uspto.gov

APPLICATION NUMBER	FILING DATE	GRP ART UNIT	FIL FEE REC'D	ATTY.DOCKET.NO	DRAWINGS	TOT CLAIMS	IND CLAIMS
09/945,176	09/04/2001	2151	355		5	7	7

CONFIRMATION NO. 3235

FILING RECEIPT

OC000000006905502

Donald Sidney Merrill
Apartment 41
294 Rio Lindo Ave.
Chico, CA 95926-5507

Date Mailed: 10/15/2001

Receipt is acknowledged of this nonprovisional Patent Application. It will be considered in its order and you will be notified as to the results of the examination. Be sure to provide the U.S. APPLICATION NUMBER, FILING DATE, NAME OF APPLICANT, and TITLE OF INVENTION when inquiring about this application. Fees transmitted by check or draft are subject to collection. Please verify the accuracy of the data presented on this receipt. **If an error is noted on this Filing Receipt, please write to the Office of Initial Patent Examination's Customer Service Center. Please provide a copy of this Filing Receipt with the changes noted thereon. If you received a "Notice to File Missing Parts" for this application, please submit any corrections to this Filing Receipt with your reply to the Notice. When the USPTO processes the reply to the Notice, the USPTO will generate another Filing Receipt incorporating the requested corrections (if appropriate).**

Applicant(s)

Donald Sidney Merrill, Chico, CA;

Domestic Priority data as claimed by applicant

Foreign Applications

If Required, Foreign Filing License Granted 10/13/2001

Projected Publication Date: 03/06/2003

Non-Publication Request: No

Early Publication Request: No

**** SMALL ENTITY ****

Title

Electronic facilitation venue

Preliminary Class

709

LICENSE FOR FOREIGN FILING UNDER
Title 35, United States Code, Section 184
Title 37, Code of Federal Regulations, 5.11 & 5.15

GRANTED

The applicant has been granted a license under 35 U.S.C. 184, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" followed by a date appears on this form. Such licenses are issued in all applications where the conditions for issuance of a license have been met, regardless of whether or not a license may be required as set forth in 37 CFR 5.15. The scope and limitations of this license are set forth in 37 CFR 5.15(a) unless an earlier license has been issued under 37 CFR 5.15(b). The license is subject to revocation upon written notification. The date indicated is the effective date of the license, unless an earlier license of similar scope has been granted under 37 CFR 5.13 or 5.14.

This license is to be retained by the licensee and may be used at any time on or after the effective date thereof unless it is revoked. This license is automatically transferred to any related applications(s) filed under 37 CFR 1.53(d). This license is not retroactive.

The grant of a license does not in any way lessen the responsibility of a licensee for the security of the subject matter as imposed by any Government contract or the provisions of existing laws relating to espionage and the national security or the export of technical data. Licensees should apprise themselves of current regulations especially with respect to certain countries, of other agencies, particularly the Office of Defense Trade Controls, Department of State (with respect to Arms, Munitions and Implements of War (22 CFR 121-128)); the Office of Export Administration, Department of Commerce (15 CFR 370.10 (j)); the Office of Foreign Assets Control, Department of Treasury (31 CFR Parts 500+) and the Department of Energy.

NOT GRANTED

No license under 35 U.S.C. 184 has been granted at this time, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" DOES NOT appear on this form. Applicant may still petition for a license under 37 CFR 5.12, if a license is desired before the expiration of 6 months from the filing date of the application. If 6 months has lapsed from the filing date of this application and the licensee has not received any indication of a secrecy order under 35 U.S.C. 181, the licensee may foreign file the application pursuant to 37 CFR 5.15(b).

Best Available Copy

DD

PTO/SB/95 (05-1999)
Approved for use through 5/31/2002 OMB 0651-0030
Patent and Trademark Office, U S

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it di

Disclosure Document Deposit Request

DISCLOSURE DOCUMENT NO.
495792
RETAINED FOR 2 YEARS
THIS IS NOT A PATENT APPLICATION
PTO-1652 (8/99)

Mail to:
Box DD
Assistant Commissioner for Patents
Washington, DC 20231

Inventor(s) Donald S. Merrill

Title of Invention Electronic Facilitation Venue

Enclosed is a disclosure of the above-titled invention consisting of 10 sheets of description and 4 sheets of drawings A check or money order in the amount of $ 10.00 is enclosed to cover the fee (37 CFR 1 21(c))

The undersigned, being a named inventor of the disclosed invention, requests that the enclosed papers be accepted under the Disclosure Document Program, and that they be preserved for a period of two years

Donald S. Merrill
Signature of Inventor

294 Rio Lindo Ave.
Address

Donald Sidney Merrill
Typed or printed name

Apartment 41

6/14/02
Date

Chico, California 95926-5507
City, State, Zip

NOTICE TO INVENTORS

It should be clearly understood that a Disclosure Document is not a patent application, nor will its receipt date in any way become the effective filing date of a later filed patent application A Disclosure Document may be relied upon only as evidence of conception of an invention and a patent application should be diligently filed if patent protection is desired

Your Disclosure Document will be retained for two years after the date it was received by the Patent and Trademark Office (PTO) and will be destroyed thereafter unless it is referred to in a related patent application filed within the two-year period The DisclosureDocument may be referred to by way of a letter of transmittal in a new patent application or by a separate letter filed in a pending application Unless it is desired to have the PTO retain the Disclosure Document beyond the two-year period, it is not required that it be referred to in the patent application

The two-year retention period should not be considered to be a "grace period" during which the inventor can wait to file his/her patent application without possible loss of benefits It must be recognized that in establishing priority of invention an affidavit or testimony referring to a Disclosure Document must usually also establish diligence in completing the invention or in filing the patent application since the filing of the Disclosure Document

If you are not familiar with what is considered to be "diligence in completing the invention" or "reduction to practice" underthe patent law or if you have other questions about patent matters, you are advised to consult with an attorney or agent registered to practice before the PTO The publication, *Attorneys and Agents Registered to Practice Before the United States Patent and Trademark Office* is available from the Superintendent of Documents, Washington, DC 20402 Patent attorneys and agents are also listed in the telephone directory of most major cities Also, many large cities have associations of patent attorneys which may be consulted

You are also reminded that any public use or sale in the United States or publication of your invention anywhere in the world more than one year prior to the filing of a patent application on that invention will prohibit the granting of a patent on it

Disclosures of inventions which have been understood and witnessed by persons and/or notarized are other examples of evidence which may also be used to establish priority

...ns of patents and patent-related ...s such as *General Information* ...consult the complete listing of all ...3) 308-HELP/4357 To ensure ...ollections, services, and hours

...osure Document Deposit Requests ...data, including gathering, preparing, ...individual case Any comments on ... the Chief Information Officer, U S ...R COMPLETED FORMS TO THIS

DONALD S. MERRILL
294 RIO LINDO AVE NO 41
CHICO, CA 95926

1031

90-3504/1211

June 14, 01 Date

Pay to the Order of Assistant Commissioner for Patents $ 10.00

Ten Dollars

TRI COUNTIES BANK
1-800-922-8742

For Disclosure Document Fee

Donald S. Merrill

⑆121135045⑆1031 0380611252⑈

APPLICATION NUMBER	FILING DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO.
09/945,176	09/04/2001	Donald Sidney Merrill	

CONFIRMATION NO. 3235

OC000000009605963

Donald Sidney Merrill
Apartment 41
294 Rio Lindo Ave.
Chico, CA 95926-5507

Title: Electronic facilitation venue

Publication No. US-2003-0046373-A1
Publication Date: 03/06/2003

Date Mailed: 03/06/2003

NOTICE OF PUBLICATION OF APPLICATION

The above-identified application will be electronically published as a patent application publication pursuant to 37 CFR 1.211, et seq. The patent application publication number and publication date are set forth above.

The publication may be accessed through the USPTO's publically available Searchable Databases via the Internet at www.uspto.gov. The direct link to access the publication is currently http://www.uspto.gov/patft/.

The publication process established by the Office does not provide for mailing a copy of the publication to applicant. A copy of the publication may be obtained from the Office upon payment of the appropriate fee set forth in 37 CFR 1.19(a)(1). Orders for copies of patent application publications are handled by the USPTO's Office of Public Records. The Office of Public Records can be reached by telephone at (703) 308-9726 or (800) 972-6382, by facsimile at (703) 305-8759, by mail addressed to the United States Patent and Trademark Office, Office of Public Records, Crystal Gateway 4, Room 335, Washington, D.C. 20231, or via the Internet.

In addition, information on the status of the application, including the mailing date of Office actions and the dates of receipt of correspondence filed in the Office, may also be accessed via the Internet through the Patent Electronic Business Center at www.uspto.gov using the public side of the Patent Application Information and Retrieval (PAIR) system. The direct link to access this status information is currently http://pair.uspto.gov/. Prior to publication, such status information is confidential and may only be obtained by applicant using the private side of PAIR.

Further assistance in electronically accessing the publication, or about PAIR, is available by calling the Patent Electronic Business Center at (703) 305-3028.

Customer Service Center
Initial Patent Examination Division (703) 308-1202

(703) 305-8497
~~John~~ Jon Lachel 703 605 4285 Publication

PTO/SB/17 (11-00)
Approved for use through 10/31/2002 OMB 0651-0032
U S Patent and Trademark Office, U S DEPARTMENT OF COMMERCE

FEE TRANSMITTAL for FY 2001

Patent fees are subject to annual revision

TOTAL AMOUNT OF PAYMENT ($) 355

Complete If Known	
Application Number	
Filing Date	
First Named Inventor	Donald Sidney Merrill
Examiner Name	
Group Art Unit	
Attorney Docket No	

METHOD OF PAYMENT

1. ☐ The Commissioner is hereby authorized to charge indicated fees and credit any overpayments to

Deposit Account Number:

Deposit Account Name:

☐ Charge Any Additional Fee Required Under 37 CFR 1 16 and 1 17

☐ Applicant claims small entity status See 37 CFR 1 27

2. ☒ **Payment Enclosed:**

☒ Check ☐ Credit card ☐ Money Order ☐ Other

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
101	710	201	355	Utility filing fee	355
106	320	206	160	Design filing fee	
107	490	207	245	Plant filing fee	
108	710	208	355	Reissue filing fee	
114	150	214	75	Provisional filing fee	

SUBTOTAL (1) ($) 355

2. EXTRA CLAIM FEES

			Extra Claims		Fee from below		Fee Paid
Total Claims	7	-20** =	0	X		=	0
Independent Claims		- 3** =		X		=	
Multiple Dependent						=	

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description
103	18	203	9	Claims in excess of 20
102	80	202	40	Independent claims in excess of 3
104	270	204	135	Multiple dependent claim, if not paid
109	80	209	40	** Reissue independent claims over original patent
110	18	210	9	** Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) ($)

**or number previously paid, if greater, For Reissues, see above

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
105	130	205	65	Surcharge - late filing fee or oath	
127	50	227	25	Surcharge - late provisional filing fee or cover sheet	
139	130	139	130	Non-English specification	
147	2,520	147	2,520	For filing a request for *ex parte* reexamination	
112	920*	112	920*	Requesting publication of SIR prior to Examiner action	
113	1,840*	113	1,840*	Requesting publication of SIR after Examiner action	
115	110	215	55	Extension for reply within first month	
116	390	216	195	Extension for reply within second month	
117	890	217	445	Extension for reply within third month	
118	1,390	218	695	Extension for reply within fourth month	
128	1,890	228	945	Extension for reply within fifth month	
119	310	219	155	Notice of Appeal	
120	310	220	155	Filing a brief in support of an appeal	
121	270	221	135	Request for oral hearing	
138	1,510	138	1,510	Petition to institute a public use proceeding	
140	110	240	55	Petition to revive - unavoidable	
141	1,240	241	620	Petition to revive - unintentional	
142	1,240	242	620	Utility issue fee (or reissue)	
143	440	243	220	Design issue fee	
144	600	244	300	Plant issue fee	
122	130	122	130	Petitions to the Commissioner	
123	50	123	50	Processing fee under 37 CFR 1 17(q)	
126	180	126	180	Submission of Information Disclosure Stmt	
581	40	581	40	Recording each patent assignment per property (times number of properties)	
146	710	246	355	Filing a submission after final rejection (37 CFR § 1 129(a))	
149	710	249	355	For each additional invention to be examined (37 CFR § 1 129(b))	
179	710	279	355	Request for Continued Examination (RCE)	
169	900	169	900	Request for expedited examination of a design application	

Other fee (specify) ____________

*Reduced by Basic Filing Fee Paid **SUBTOTAL (3)** ($)

SUBMITTED BY				Complete *(if applicable)*	
Name (Print/Type)	Donald Sidney Merrill	*Registration No (Attorney/Agent)*		*Telephone*	(530) 897-0735
Signature				*Date*	

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

ABSTRACT

Information Technology offers exciting opportunities for an organization or individual to reduce costs and achieve limitless productivity increases. However, if these gifts are to be fully realized and sustained, we must understand that the widespread use of Information Technology must inevitably produce problems as well as benefits. The first step in dealing with any such problems is to recognize that they, in fact, exist. Current dislocations appearing in all sectors of society gives inescapable proof of major problems and forced us to question our present methodology. But these factors alone can not delineate the problems nor answer the big question. What can be done to solve them? These problems and the questions they have generated are the subject of this discussion. Its purpose is to define what the problems are, then offer a solution.

In its beginnings, communication required face-to-face interaction so these exchanges took place locally where people congregated. Since both parties lived close together, knew and depended on the other for survival, the cost for such communication was small. As local venues gave way to regional centers and port cities, improvements in transportation and communication brought problems of time and space to the fore. Resulting in a critical lack of information.

While some individuals did attempt to operate in these new venues, time and distances made it difficult to set the conditions that balance supply and demand. A system of specialized transaction methods appeared to fill the void created by the factors of Space/Time. Designed to facilitate communication, they all became essential to a growing world population. In their own way, each made it possible to get around the limitations imposed. If these factors are removed and their

limitations are no longer present a facilitation venue capable of exploiting this new environment is mandatory.

About the mid-20th century (IT) Information Technology's influence on these factors became so profound it created a change in the entire fabric of the world's social and economic structure. In turn, this same technology created such serious problems in their practical application that it threatened to bring a halt to all future development. To prevent this would require new ways of thinking about how and where IT operates.

In the era of mainframe computing, the questions of venue, facilitation, and security were integral parts of centralized computing. A "clean", steady-state-power supply, cooled and filtered air, large well-trained staff were required to maintain the continuous operation of the system. A centralized IT department was needed to support and dispense corporate information and a natural wall security created by limited access developed around the system.

The Interorganizational IT environments envisaged by Felix Kaufman in his article, 1966 Harvard Business Review, have no single defining venue or environment. System's components are distributed through out a network of remote locations. Security was accomplished by specialized hardware and software. System costs are determined by the degree of insecurity you were willing to accept. Complete security tended to make the system impossible to use.

Finally, by mid 1980s and early 90s of the 20th Century organizations had no choice; extend relationships to the Internet or be at a competitive disadvantage. But such decisions made under market pressure forced organizations to accept operational trade-offs that gave time-to-market priority over internal-system-continuity and operational efficiency. They had to execute transaction strategies in a technical and operational vacuum. They did not have the time to integrate new Internet Protocols with their legacy systems, and design processes, databases, and organizational structures to manage this new way of conducting transactions in concert with their traditional organization.

Sometimes, disconnect was due to pressures from external forces, at other times it was because executive management was skeptical of the long-term impact of this new form. Whatever the reasons, most companies that implemented these initiatives created disconnect organizations that they could launch in the face of time pressures, organizational discord, and management uncertainty. The biggest problem, they lost money, and now there was another challenge. Continue delivering value via the Internet and your other channels while still making a profit. To survive the aftermath of doing business on the Internet organizations had to produce revenue.

The Internet was the first true venue for Information Technology, and as we will indicate, very successful within the context of its intended function. It was never intended to be a complete answer. What is required is an Electronic Facilitation Venue that is architecturally flexible enough to successfully facilitate any IT function but robust enough to establish itself as a distinct system.

If this discussion is to use terms like Facilitation and Venue, we must first offer definitions within the context of this document. For purposes of the discussion we shall use the term Venue as a place of location, either physical or virtual, where a function takes place.

Examples of physical venues are: Court Buildings, where the function is the execution of criminal and civil trials. Concert Halls and baseball Parks etc., where the name itself indicates their function. Time and space has always imposed limits on the physical location. However, this paper will focus on the virtual or electronic venue; a form made possible by the advent of Information Technology (IT).

The term Facilitation is used to indicate an activity or process "that lowers the threshold" for an activity to take place. Information Technology tends to eliminate time and space, therefore a new kind of facilitation is required. We shall show that the Electronic Facilitation Venue is the answer to the problems

encountered by business in their attempts to use the Internet for activities its architecture was never intended to support.

One other point is germane to this discussion. The Internet is not the soul problem in the application of Internet Technology. The failure of the Retail and Manufacturing sectors of the economy to produce sufficient profits in this new economy brought about excessive monetary accommodation by central banks and the misallocation of trillions of dollars that destabilized the global economy. Arrogant financial market participants supported by new financial tools such as state-of-the-art computers and telecommunication, securitization and derivatives generated a belief that the usual guides to sound finance and investment had become 'old-fashioned'. Eventually, the global market collapsed into recession from the contagion of incompetence and greed; however, it happened because Information Technology provided the tools that made it all possible.

SPECIFICATIONS

Title and Background

None of the examples presently represented by organizations doing business on the Internet are free from the problems of the Internet. Security is dependent upon the cost and degree of risk the vendor and consumer are willing to accept. Yet the answer is simple, secure business can not be done on the internet but to change or modify it would be prohibitively expensive. Therefore, a system must be developed specifically for business yet capable of using the Internet for purposes it was designed to provide. It is called the Electronic Facilitation Venue.

Components

The Electronic Facilitation Venue (EFV) (see page 13, fig. 1) consists of six components: The "Host" made up of a Processor Pool + Central Processing Unit (CPU). Specialized Servers for files, database, and clients. Data Warehouse where data is analyzed summarized then turned into useful information. An Ethernet Backbone designed to provide wide band, high-speed communication between system components. Routers assigned to a specific EFV use advanced encryption and tunneling to permit secure, invisible communication on the Internet. An Internet Service Access (ISA) handles incoming Internet packets where it is checked by a Router using packet filtering, then handled by proxy services acting as agents for clients. A Bastion Host receives the results from the proxy services where its dual network cards effectively cuts any direct link to the EFV thus bringing to a dead end any Internet communication by an unauthorized user; a screening Router attached to the second network card eliminates any traffic not identified as LEFV Host traffic. The system requires one additional component that can not be shown in any diagram. A Distributed Operating System (DOS).

Current Examples

Internet2 is a not-for-profit networking consortium led by the academic and research communities. It operates a next-generation Internet Protocol and optical network the delivers production network services to the high-performance demands of research and education. ***Web2.0*** is an Internet web-development and web-design system that facilitates interactive information sharing, interoperability, user-centered design and collaboration. ***Cloud Computing*** is computing in which scalable and virtualized resources are provided as a service. Users need not have knowledge of, or control over the technology that supports them. The term cloud is a metaphor for the Internet as it is depicted in computer network diagrams an abstraction of the complex infrastructure it conceals.

Conclustions

The initial focus must be on establishing and enforcing limited access by Internet traffic. To do this requires isolation of the system from the Internet, but it must have full access to the Internet to function as an Internet business. In the 40's and 50's of the Twentieth Century the U.S. Department of Defense (DOD) wanted to network its many different mainframe computers to achieve secure and reliable exchange of information. Using Packet Switching Technology (PST), Transport Control Protocol (TCP), and the Internet Protocol (IP), they created the first large scale deployment of the Internet. A great deal of time and effort has been invested in the development of the Internet's architecture to create an efficient venue for exchange of information; however, this very specific function was never designed to do anything else. These facts are central to understanding the subject of this proposal: The Electronic Facilitation Venue (EFV).

Today, all user problems on the Internet are found in designs centered on solutions that involve using off-the-shelf software and hardware to create a webpage where Information Technology (IT) can operate without the necessity of

imposing security features that increase costs and restrict system performance. In order to implement an entity that would conform to the outline above requires a very special type of Venue, and the application of highly sophisticated network technology. It will require a new way of looking at the problem and a brand new perspective on its implementation. What follows will demonstrate how the Internet is being tinkered with, manipulated, and changed in hopes it will fit the needs of business. All these efforts have failed because they have all been designed to work within the Internet Protocol that is unable to provide a venue suitable for conducting business. My patent application has the answer to the problem.

CLAIMS

I claim My Idea if accepted For Patent Protection and Implemented Will Show:

1. Virtual Space formed by construction of an electronic enclosure produces utility as a natural product, resulting in an Electronic Facilitation Venue

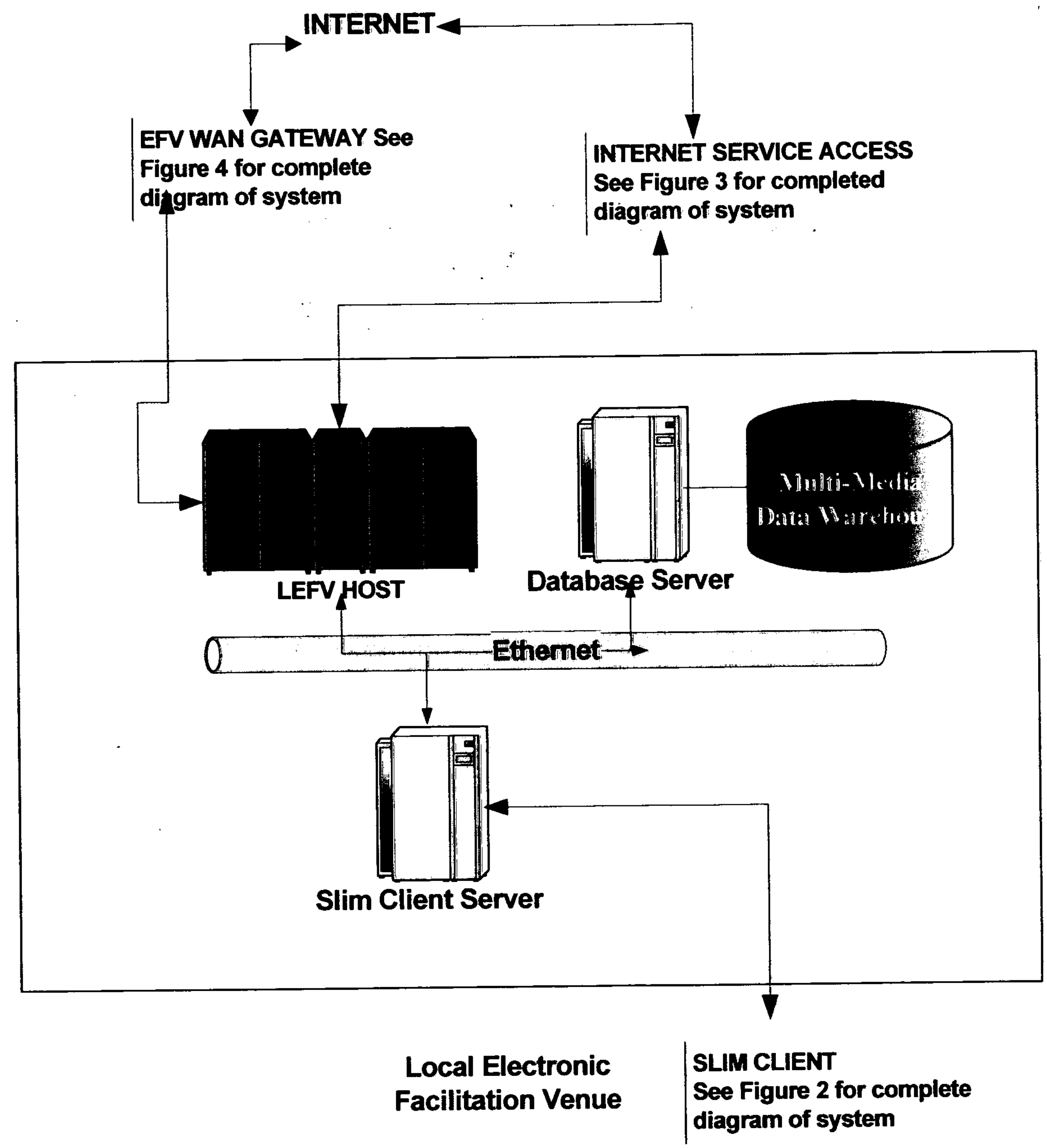
- The above terms require definitions. The term facilitation is used to indicate an activity, process, or venue that lowers the 'threshold' required for a process to take place. For purposes of this claim we shall use the term Venue as a place or location, either physical or virtual, where a process can takes place. Time, space, and dimensions limit physical locations. These limitations often create the need for extensive physical facilitation. Since Information Technology tends to eliminate these factors virtual facilitation increases productivity and enhances cost effectiveness.
- The Electronic Facilitation Venue proposed here is a 'Generic Venue' that does not require immediate, specific hardware or software definition. It is an intellectual construct made manifest by Micro-electronic engineering that produces the required physical components for its completion, (see Fig. 1, page 13). Care must be taken in determining IT type and nature of its processes because these will determine the final configuration of the Facilitation Venue.

2. It will constitute a single architecture comfortable with a wide variety of IT operations.

- The venue is neither Legacy nor Network architecture. It is an electronic enclosure constructed from off-the-shelf hardware and software. 'Thin-clients' are used to control end-user access, (see fig. 2, page 15) and a distributed operating system determines program content, types, and degree of access.
- The user interface will have a similar 'feel' to that of an electronic game, but have program content appropriate to the IT process and the end user.
- The key factor is this architecture is not focused on the Internet's 'distribution of information' but designed to offer a separate and secure venue for its exchange.

THIS SPACE INTENTIONALLY NOT USED
PLEASE TURN PAGE

Fig. 1


INTERNET
EFV WAN GATEWAY See Figure 4 for complete diagram of system
INTERNET SERVICE ACCESS See Figure 3 for completed diagram of system
LEFV HOST
Database Server
Ethernet
Slim Client Server
Local Electronic Facilitation Venue
SLIM CLIENT See Figure 2 for complete diagram of system

Local Electronic Facilitation Venue
(See opposite page, figure. 1 for Reference)

The diagram to the left represents the real purpose for and the object of this patent application. Its title is Electronic Facilitation Venue (EFV). It is Electronic because it is constructed from Micro-electronic components and only exists as an electronic entity; Facilitation because it lowers the 'threshold' required for activation and offers a secure environment for any process needing its services; Venue because it is a place where any suitable Information Technology process can take place.

The diagram shows a Local Electronic Facilitation Venue (LEFV) consisting of six components: The "Host" made up of a Processor Pool + Central Processing Unit (CPU). Specialized Servers for files, database, and clients. Data Warehouse where data is analyzed summarized then turned into useful information. Ethernet Backbone designed to provide wide band, high-speed communication between components of the EFV. The EFV Gateway is designed to be the most secure connection on the Internet and at the same time invisible to all other Internet traffic; LEFV Routers assigned to a specific LEFV uses advanced encryption and tunneling to permit secure, invisible communication on the Internet between LEFVs. The Internet Service Access (ISA) where incoming Internet packets are first checked by a Router using packet filtering. In the second step, incoming Internet traffic is handled by proxy services acting as agents for LEFV clients. A third step follows with the Bastion Host where its dual network cards effectively cuts any direct link to the LEFV thus bringing to a dead end any Internet communication by an unauthorized user; a screening Router attached to the second network card eliminates any traffic not identified as LEFV Host traffic.

This system requires one component that can not be shown in the diagram. A Distributed Operating System (DOS) based on a Microkernel + Server architecture and is designed to take a collection of machines and make them work

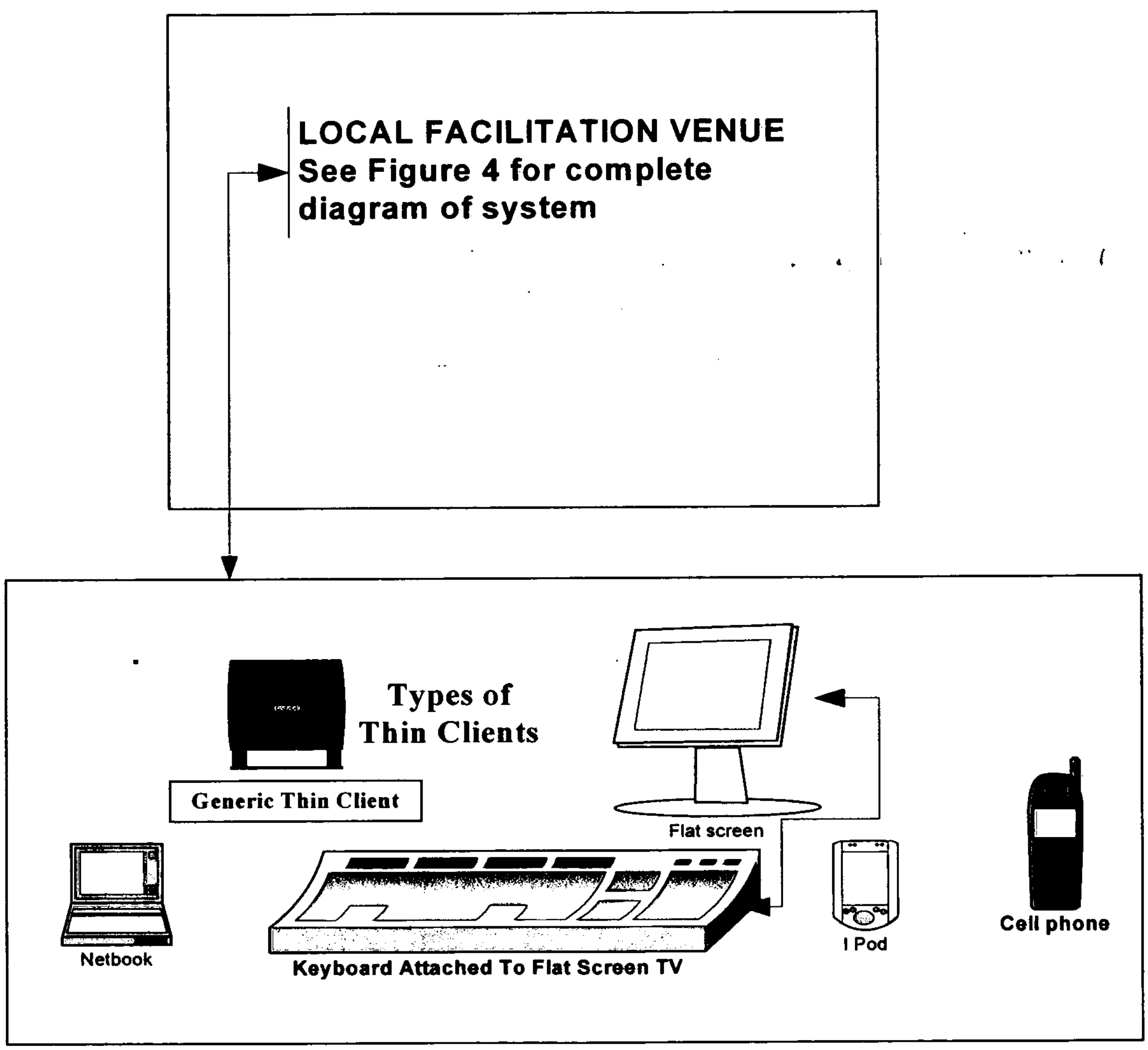
together. Every component in the LEFV runs a small, identical piece of software called a ***kernel*.** This kernel supports the basic processes, communication, and object primitives. It also handles device I/O and memory management. Thus a Local Electronic Facilitation Venue is structured so that a collection of independent machines and processes appears to be a single entity

Because of their simplicity thin clients are immune to the abuses normally associated with a more robust workstation, PC or Laptop Computer, and can be designed to deliver fast deployment of both application software and hardware with higher reliability, less cost, grater manageability, and security. The end-user of the system need only deal with a simple appliance to access any or all services. The primary function of the Electronic Facilitation Venue is the exchange of *Useful* Information. It is not a place to play games, get the sports news, nor the latest Political Klack. The entrepreneur can conduct business without interruption by criminals!

We often hear people talk of using the Internet for this or that project. This idea is a consequence of simple misconception about the nature of the Internet. As we indicated earlier, the Internet is designed to facilitate exchange of information between disparate computers. This is a "service" rather than a "utility" function. A utility function requires virtual space formed by construction of an electronic enclosure. Although this space is an intellectual construct, the enclosure formed is real and must be carefully designed to give maximum freedom to the user ***inside***, but none at all to those on the ***outside***.

THIS SPACE INTENTIONALLY NOT USED
PLEASE TURN PAGE

Fig. 2



EFV THIN CLIENT USER

Thin Client System

(See opposite Page, Figure 2 for reference)

Thin client is a generic term used to describe an appliance designed to execute application software received from a client server based on the LEFV. The appliance receives application software and displays it to the client. Because of their relative simplicity, thin clients can be designed to deliver higher reliability, as well as much easier manageability, with faster deployment of both software and hardware, at far lower cost than you would get with a personal computer. Thin clients can be as large as a "dumb" terminal or as small as a hand calculator.

Thin client immunity to problems presently seen with Internet use is based in part by the fact it does not have the complexities of PCs, Laptops or Workstations. A study determined 70% of viruses were introduced through floppy disks, the others come imbedded in applications designed to attack the hard drive, where the file application tables are erased; thus, the drive no longer knows where the data is stored. By eliminating this hardware, thin clients avoid viruses and other invasive constructs.

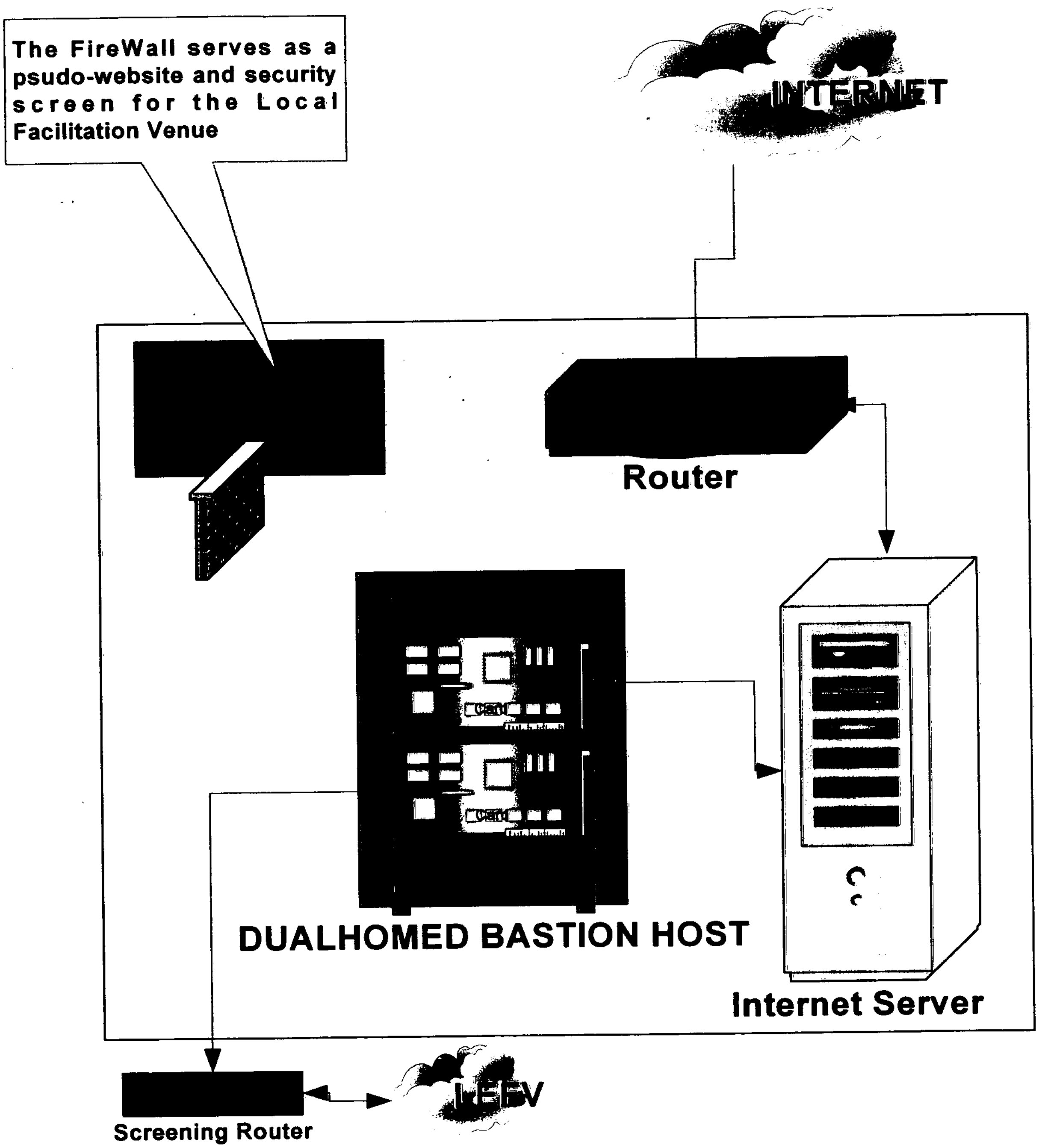
The Netbook is an example of such an appliance and is about the size of a laptop computer. It is designed for wireless communication or access to the Internet. Consumers are targeted by "Cloud" and "Internet 2" Service providers who require a less powerful appliance. Application software and virtualized resources are provided as a service. The Netbook's firmware provides the intelligence needed to receive application software, but its simplified architecture make the devices substantially smaller, cheaper, and easier to use than the typical workstation or personal computer. The power and control of the personal computer or the "Smart" terminal is no longer needed. In addition, the new wireless appliances such as smart phones and portable media players can be

designed to distribute services offered by LEFV. Their simplicity can reduce the cost so that they might be offered free to the client in turn for accepting services.

As was indicated earlier, it is important to prevent unauthorized access to the LEFV from outside sources. By eliminating the hardware that characterizes the Smart Workstation and the Personal computer with specialized Client Servers driven by a Distributed Operation system and investing the thin client with a "Kernel" answerable only to the LEFV we have effectively eliminated most of the opportunity for unauthorized access. In addition we have isolated the EFV from Internet traffic that often carries the opportunist to vulnerable targets.

THIS SPACE INTENTIONALLY NOT USED PLEASE TURN PAGE

Fig. 3



Internet Service Access

Internet Services Access (ISA)

(See opposite Page, Figure 3 for reference)

Three types of hardware handle LEFV second Internet contact: two Routers, Internet Server, and a Dual Homed Bastion Host. These hardware components make up what is known as a "Firewall". Hardware makes implementation possible while application software acts as the process director. The diagram shows a generic Host with two network cards as our first line of real security for traffic from the Internet. Many computer systems have the ability to function with more than one network card. Separate cards effectively cut the direct link and isolates incoming Internet traffic from the LEFV. An Internet Server acts as a gatekeeper and proxy to analyze all in coming traffic destined for the Bastion Host.

Incoming Internet packets are first checked by a Router using packet filtering then either dropped or allowed to enter based on various rules and specified criteria. In the second step, proxy services act as agents for EFV clients that the Internet user needs to communicate with on the other side of the firewall. There are two advantages of proxy servers. First, users do not directly control requests for access nor do they log onto or have an account on the Bastion Host. Second, the use of audit trails allows the server to keep track of the type and number of the transactions on the server. The Bastion Host with its dual network cards effectively cuts any direct link to the LEFV thus becomes a dead end for any Internet communication by an unauthorized user. A screening Router attached to the second network card of the Bastion eliminates any traffic not identified as LEFV Host traffic.

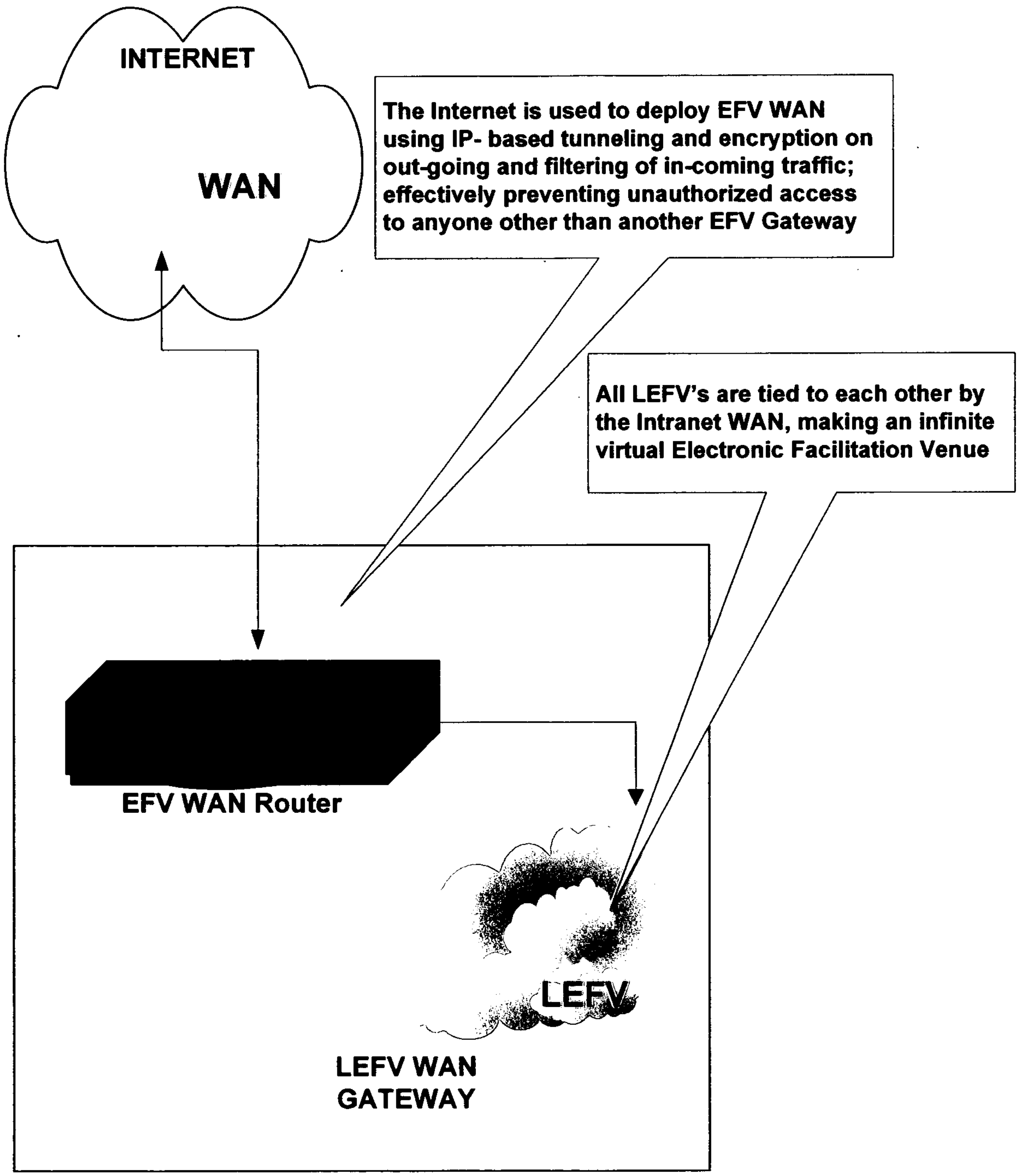
As is indicated by the diagram traffic on ISA is two-way. Traffic from the LEFV must traverse security procedures so that the proxy server can process out-going traffic. The firewall proxy server on a particular LEFV possesses the only Internet address for that LEFV.

The goal is to perform the security function by cutting any direct link to the EFV and enhancing it through multiple layers of defense. This reduces the need for passwords, codes, and dependence upon the human factor. In addition, the proxy server produces audit trails that can spot traffic that might cause future problems and thus blocked from reentry. A concern often voiced by consumers is the lack of control over personal information when using the type of service presented here. This concern can be addressed by supplying a secure off-site storage for clients that request or need such services.

It is important for the examiner of this patent proposal to understand that the ISA Firewall represents a significant change from the current style of business on the Internet. Vendors practicing Cloud Computing, operating in Web 2.0, and Internet 2 all use the cloud a symbol to depict the Internet where resources are provided as a service. The consumer (client) must use the Internet to find and receive services. All of the security problems of the Internet apply.

THIS SPACE INTENTIONALLY NOT USED
PLEASE TURN PAGE

Fig. 4


INTERNET
WAN
The Internet is used to deploy EFV WAN using IP- based tunneling and encryption on out-going and filtering of in-coming traffic; effectively preventing unauthorized access to anyone other than another EFV Gateway
All LEFV's are tied to each other by the Intranet WAN, making an infinite virtual Electronic Facilitation Venue
EFV WAN Router
LEFV
LEFV WAN GATEWAY

Local Electronic Facilitation Venue (LEFV) WAN Gateway

(See opposite page, Figure 4 for Reference)

The diagram represents the component parts of the local connection to the Internet by the LEFV. It is designed to be the most secure connection for communications on the Internet and at the same time must be transparent to other Internet traffic. Each LEFV Router assigned to a specific LEFV uses advanced encryption and tunneling to permit the organization to establish secure LEFV communications.

Gateway filters must limit incoming traffic to the specific address of the LEFV. To do this, requires a unified solution of a more robust Router and layer 3 IP-based tunneling and encryption. In addition, it will require single port deployment to an IP cloud to attain meshed connectivity among each location on the Intranet WAN.

The Cisco 7100 series Router is an example of such a unit. "Cisco 7100 Series Router (Cisco Systems, Inc.) deliver tunneling and encryption services suitable for sit-to-site Intranet, Extranet, and applications. As scalability requirements increase, an optional Integrated Services Adapter is installed for encryption acceleration and tunnel scalability. For perimeter security applications, the 1700 also support IOS Firewall feature sets, enabling packet filtering on the routing infrastructure. This system enables the enterprise to choose WAN transport best suited to their needs." This example of off-the-shelf equipment available for specific tasks may require some modification but are minor and cost effective.

The central idea of this system is the elimination of any direct link to the LEFV by Internet traffic. At the same time it establishes a presence on the Internet for secure reception of Intranet traffic for an Extranet entity. This allows the Internet to do what it does best: connecting Extranet operations, and even

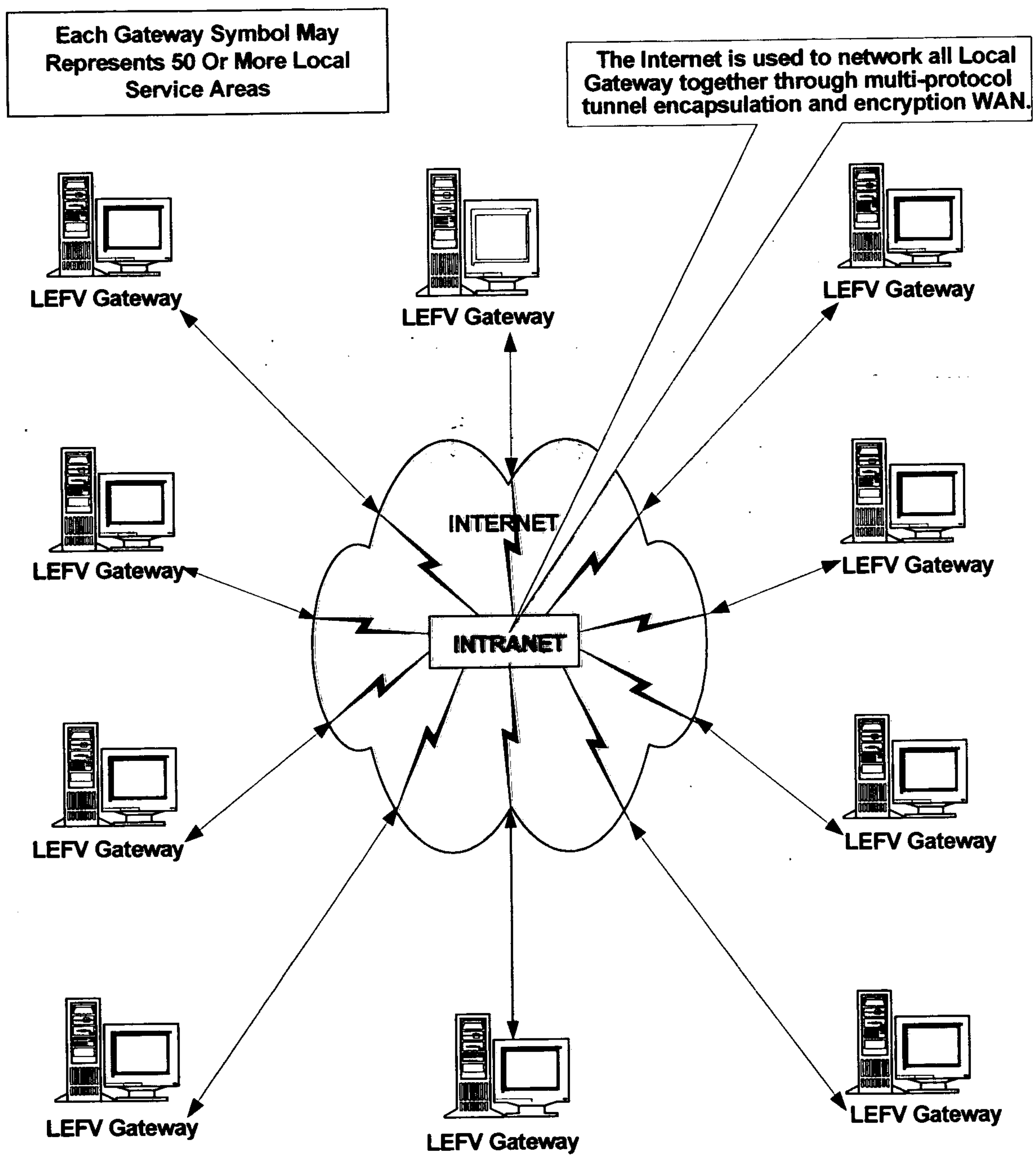
delivering information worldwide. The fact that the Internet is presently international in scope establishes the possibility for expansion of the EFV. The use of the Internet can provide secure connection for communication to each LEFVs as the system expands! It acts as an enhancement to the Internet rather than being simply a user. Both parties can profit by the partnership, the Internet can now do what it does best and the EFV gives the entrepreneur a scalable, elastic, and safe place to conduct business. The partnership will create a Twenty First-Century Communication Network.

There has been a concerted effort by many companies to develop a viable computing infrastructure that would deliver services with different levels of virtualization technologies. All have met with varying degrees of success; however, all conduct their operations using the Internet.

What is Internet 2.0? Contenders such as Web 2.0, Internet 2, and Cloud Computing all are part of the Internet, as a result they are subject to the problem of the Internet. The EFV enhances but is not part of the Internet and uses it only as a conduit for communications. Thus can legitimately be considered as a candidate for the title "Internet 2.0".

THIS SPACE INTENTIONALLY
NOT USED
PLEASE TURN PAGE

Fig. 5



Electronic Facilitation Venue WAN

Internet Communication of Local Electronic Facilitation Venue (LEFVs)

(See: figure 5, opposite page for reference)

The Internet has made it possible for people to effectively and inexpensively communicate with one another. Unlike traditional media, the Internet does not have a centralized distribution system. Instead, an individual with Internet access can communicate directly with anyone else on the Internet, make information available to others, find information provided by others, or sell products with a minimum overhead cost.

The Problems of Technological Change

The economy has tanked due to a broken promises by technology! An overheated market has crashed, turning heroes of business into goats and scoundrels. Disillusionment reigns, and no one knows what's going to happen next.

Sound familiar? Perhaps not! The time of which we speak is 1850! The place is not the United States but England. The new technology is steam, and hundreds of companies have gone belly-up. What happened next is even more fascinating. After the bust, the economy leveled off and then resumed a steady climb. By 1870 the economy was back on track. The bust had cleared the market of speculative stock swindles, leaving only solid companies.

According to economic research the same pattern holds for three other tech-driven economic movements as well: the Industrial Revolution of the late 1700s in England, the age of cheap steel and electricity in the late 1800s in the U.S. and Germany, and the automobile and mass-production era starting about 1910. After a gestation period of a decade or more, new technology sparks a boom followed by a sudden bust, leading to widespread confusion. The key to seizing opportunities offered by new technology is patience. Despite the early rapid rise and collapse of the economy, the full cycle can take at least 50 years to play out from breakthrough to maturity. During these years misapplication of the new technology produces a lot of chaos and pain. The last half of the Twentieth Century brought a Global Recession.

New Opportunities

The arrival of the Twenty First Century marked fifty years since Information Technology was first discovered. As the century moved into its first decade, the new technology begun to show signs of maturity. Open-source software has begin to improve in quality and versatility, companies have found ways to make money by providing services to open-source-software users, or by packaging this free software with products they sell. Utility computing" where a company provides you with computing power when you need it. Rather than buying hardware and software yourself, you would pay companies for the work delivered -- and it has some big backers.

As we look forward change is inevitable, but the internal architecture of the Internet can not accommodate business activity. Does this mean that business must now abandon this best hope for future expansion of twenty first century business. The answer is based on the logical precept that simplicity is best. The Internet's architecture is an open, decentralized network, placing the intelligent components directly into end-user hands. It is the ideal model for innovation and patent application 09/945,176 will take advantage of this.

Each icon in fig. 5, opposite page 21 represents Local Facilitation V*enue* Gateways connected to the Internet by a 'Router'. This is a generic designation for this piece of hardware since they are most frequently used to connect two logically and physically different networks; however, in this case each will function as a secure link between identical LEFVs. Each must be able to send and receive traffic only between LEFVs and designed to create advanced encryption and tunneling communication of its out-going traffic. In-coming traffic for each Gateway will be processed and filtered so that only the specific address of each Gateway is allowed access. As a result LEFV traffic will be virtually invisible and inaccessible to other Internet traffic.

The EFV is outside and totally separate from the Internet. It is a new entity designed to provide a secure venue for Information Technology. An open,

decentralized architecture used by a new, complimentary system dedicated to placing intelligent components directly into the hands of the end-user and involves provision for dynamically scalable and often virtualized resources as a service. We are beginning to hear the term Internet 2.0 but all present offerings represent organizations much like Wire news services, stock ticker tape, wire funds transfers, intercontinental cables, that preceded the Telephone. The Electronic Facilitation Venue is Internet 2.0!

OIPE IAP40
JAN 04 2010
PATENT & TRADEMARK OFFICE

Notice of Abandonment	**Application No.** 09/945,176	**Applicant(s)** MERRILL, DONALD SIDNEY	
	Examiner David Y. Jung	**Art Unit** 2134	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

This application is abandoned in view of:

1. ☒ Applicant's failure to timely file a proper reply to the Office letter mailed on _7/18/2007_.
   (a) ☐ A reply was received on _____ (with a Certificate of Mailing or Transmission dated _____), which is after the expiration of the period for reply (including a total extension of time of _____ month(s)) which expired on _____.
   (b) ☐ A proposed reply was received on _____, but it does not constitute a proper reply under 37 CFR 1.113 (a) to the final rejection.
   (A proper reply under 37 CFR 1.113 to a final rejection consists only of: (1) a timely filed amendment which places the application in condition for allowance; (2) a timely filed Notice of Appeal (with appeal fee); or (3) a timely filed Request for Continued Examination (RCE) in compliance with 37 CFR 1.114).
   (c) ☐ A reply was received on _____ but it does not constitute a proper reply, or a bona fide attempt at a proper reply, to the non-final rejection. See 37 CFR 1.85(a) and 1.111. (See explanation in box 7 below).
   (d) ☒ No reply has been received.

2. ☐ Applicant's failure to timely pay the required issue fee and publication fee, if applicable, within the statutory period of three months from the mailing date of the Notice of Allowance (PTOL-85).
   (a) ☐ The issue fee and publication fee, if applicable, was received on _____ (with a Certificate of Mailing or Transmission dated _____), which is after the expiration of the statutory period for payment of the issue fee (and publication fee) set in the Notice of Allowance (PTOL-85).
   (b) ☐ The submitted fee of $_____ is insufficient. A balance of $_____ is due.
   The issue fee required by 37 CFR 1.18 is $_____. The publication fee, if required by 37 CFR 1.18(d), is $_____.
   (c) ☐ The issue fee and publication fee, if applicable, has not been received.

3. ☐ Applicant's failure to timely file corrected drawings as required by, and within the three-month period set in, the Notice of Allowability (PTO-37).
   (a) ☐ Proposed corrected drawings were received on _____ (with a Certificate of Mailing or Transmission dated _____), which is after the expiration of the period for reply.
   (b) ☐ No corrected drawings have been received.

4. ☐ The letter of express abandonment which is signed by the attorney or agent of record, the assignee of the entire interest, or all of the applicants.

5. ☐ The letter of express abandonment which is signed by an attorney or agent (acting in a representative capacity under 37 CFR 1.34(a)) upon the filing of a continuing application.

6. ☐ The decision by the Board of Patent Appeals and Interference rendered on _____ and because the period for seeking court review of the decision has expired and there are no allowed claims.

7. ☐ The reason(s) below:

Primary David Jung
Art Unit 2134

2/4/08

Petitions to revive under 37 CFR 1.137(a) or (b), or requests to withdraw the holding of abandonment under 37 CFR 1.181, should be promptly filed to minimize any negative effects on patent term.

United States Patent and Trademark Office

Patent Application Information Retrieval

Select Search Method: Application Number

Enter Number:

SUBMIT

Order Certified Application As Filed
Order Certified File Wrapper
View Order List

09/945,176 Electronic facilitation venue

Application Data | Transaction History | Image File Wrapper | Published Documents | Publication Dates | Address & Attorney/Agent

Date	Contents Description
02-27-2006	**Miscellaneous Incoming Letter**
12-20-2004	Miscellaneous Incoming Letter
11-30-2004	**Mail Non-Final Rejection**
11-29-2004	Non-Final Rejection
11-23-2004	**Case Docketed to Examiner in GAU**
09-17-2004	IFW TSS Processing by Tech Center Complete
08-13-2003	**Case Docketed to Examiner in GAU**
03-08-2002	Case Docketed to Examiner in GAU
01-19-2002	**Transfer Inquiry to GAU**
10-15-2001	Application Dispatched from OIPE
10-13-2001	**Correspondence Address Change**
09-13-2001	IFW Scan & PACR Auto Security Review
09-04-2001	**Initial Exam Team nn**

If you need help:

Patent Application Information Retrieval

Select Search Method: Application Number

Enter Number:

SUBMIT

Order Certified Application As Filed
Order Certified File Wrapper
View Order List

09/945,176 Electronic facilitation venue

Application Data | Transaction History | Image File Wrapper | Published Documents | Publication Dates | Address & Attorney/Agent

Application Number:	09/945,176	**Customer Number:**	-
Filing or 371 (c) Date:	09-04-2001	**Status:**	Non Final Action Mailed
Application Type:	Utility	**Status Date:**	11-30-2004
Examiner Name:	JUNG, DAVID YIUK	**Location:**	ELECTRONIC
Group Art Unit:	2134	**Location Date:**	-
Confirmation Number:	3235	**Earliest Publication No:**	US 2003-0046373 A1
Attorney Docket Number:	-	**Earliest Publication Date:**	03-06-2003
Class / Subclass:	713/201	**Patent Number:**	-
First Named Inventor:	Donald Merrill , Chico, CA (US)	**Issue Date of Patent:**	-
Title of Invention:	Electronic facilitation venue		

If you need help:

- *Call the Patent Electronic Business Center at (866) 217-9197 (toll free) or e-mail EBC@uspto.gov for specific questions about Patent Application Information Retrieval (PAIR).*
- *Send general questions about USPTO programs to the USPTO Contact Center (UCC).*
- *If you experience technical difficulties or problems with this application, please report them via e-mail to Ele Business Support or call 1 800-786-9199.*

Amoeba — A Distributed Operating System for the 1990s

Sape J. Mullender

Guido van Rossum

CWI, the Centre for Mathematics and Computer Science

Andrew S. Tanenbaum

Robbert van Renesse

Hans van Staveren

Vrije Universiteit

ABSTRACT

Amoeba is the distributed system developed at the Free University (VU) and Centre for Mathematics and Computer Science (CWI), both in Amsterdam. Throughout the project's ten-year history, a major concern of the designers has been to combine the research themes of distributed systems, such as high availability, use of parallelism and scalability, with simplicity and high performance. Distributed systems are necessarily more complicated than centralized systems, so they have a tendency to be much slower. Amoeba was always designed to be used, so it was deemed essential to achieve extremely high performance. We are working hard to achieve this goal — Amoeba is already one of the fastest distributed systems (on its class of hardware) reported so far in the scientific literature and future versions will be even faster.

The Amoeba software is based on objects. An object is a piece of data on which well-defined operations may be performed by authorized users, independent of where the user and object are located, Objects are managed by server processes and named using capabilities chosen randomly from a sparse name space.

Processes consist of a segmented address space shared by one or more threads of control. Processes can be created, managed and debugged remotely. Operations on objects are implemented using remote procedure calls.

Amoeba has a unique and fast file system. The file system is split into two parts — the Bullet Service, which stores immutable files contiguously on the disk, and the SOAP Directory Service, which provides a mechanism for giving capabilities symbolic names. The directory server also handles replication and atomicity, eliminating the need for a separate transaction management system.

To bridge the gap with existing systems, Amoeba provides a Unix emulation facility. This facility contains a library of Unix system call routines, each of which does its work by making calls to the various Amoeba server processes.

Since the original goal of the design was to build a fast system, some actual performance measurements of the current implementation are given. A remote procedure call can be performed in 1.4 msec on Sun-3/50 class machines, and the file server can deliver data continuously at a rate of 677 kbytes/sec.

1. INTRODUCTION

The 1970s were dominated by medium to large sized time-sharing systems, typically supporting 10 to 100 on-line terminals. In the 1980s, personal computing became popular, with many organizations installing large numbers of PCs and engineering workstations, usually connected by a fast local area network. In the 1990s, computer prices will drop so low that it will be economically feasible to have 10, 20, or perhaps 100 powerful microprocessors *per user*. The key issue is how to organize all this computing power in a simple, efficient, fault-tolerant, and especially, easy to use way. In this paper we describe a distributed operating system that meets this challenge.

The basic problem with current networks of PCs and workstations is that they are not *transparent*, that is, the users are conscious of the existence of multiple machines. One logs into a specific machine and uses that machine only, until one does a remote login to another machine. Few, if any, programs can take advantage of multiple CPUs, even if they are all idle, for example. An operating system for connecting a number of autonomous computers is usually called a *network operating system*.

In contrast, the kind of system we envision for the 1990s *appears* to the users as a single, 1970s centralized timesharing system. Users of this system are not aware of which processors their jobs are using (or even how many), they are not aware of where their files are stored (or how many replicated copies are being maintained to provide high availability) or how communication is taking place among the processes and machines. The whole thing just looks like a single big timesharing system. All resource management is done completely automatically by what is called a *distributed operating system*.

Few such systems have been designed, and even fewer have been implemented. Fewer still, are actually used by anyone (yet). One of the earliest distributed systems was the Cambridge Distributed Computing System [Needham and Herbert, 1982] Later, other systems were developed, such as Locus [Walker at al, 1983], Mach [Accetta et al, 1986], V-Kernel [Cheriton, 1988], and Chorus [Rozier et al., 1988]. Most of the classical distributed systems literature, however, describes work on parts of, or aspects of distributed systems. There are many papers on distributed file servers, distributed name servers, distributed transaction systems, and so on, but there are few on whole systems.

In this paper we will describe a research project — Amoeba — in which a working prototype system was successfully constructed. We will cover most of the traditional operating system design issues, including communication, protection, the file system, and process management. We will not only explain what we did, but also why we did it.

2. OVERVIEW OF AMOEBA

The Amoeba Project [Mullender and Tanenbaum, 1986] is a joint effort of groups at the Free University (VU), and the Centre for Mathematics and Computer Science (CWI), both in Amsterdam. The project has been underway for nearly ten years and has gone through numerous redesigns and reimplementations as design flaws became glaringly apparent. This paper describes the Amoeba 4.0 system, which was released in 1990.

The work described here has been supported by grants from NWO, the Netherlands Research Organization, SION, the Foundation for Computer Science Research in the Netherlands, OSF, the Open Software Foundation, and Digital Equipment Corporation.

2.1. The Amoeba Hardware Architecture

The Amoeba hardware architecture is shown in Fig. 1. It consists of four components: workstations, pool processors, specialized servers, and gateways. The *workstations* are intended to execute only processes that interact intensively with the user. The window manager, the command interpreter, editors, CAD/CAM graphical front-ends are examples of programs that might be run on workstations. The majority of applications do not usually interact much with the user and are run elsewhere.

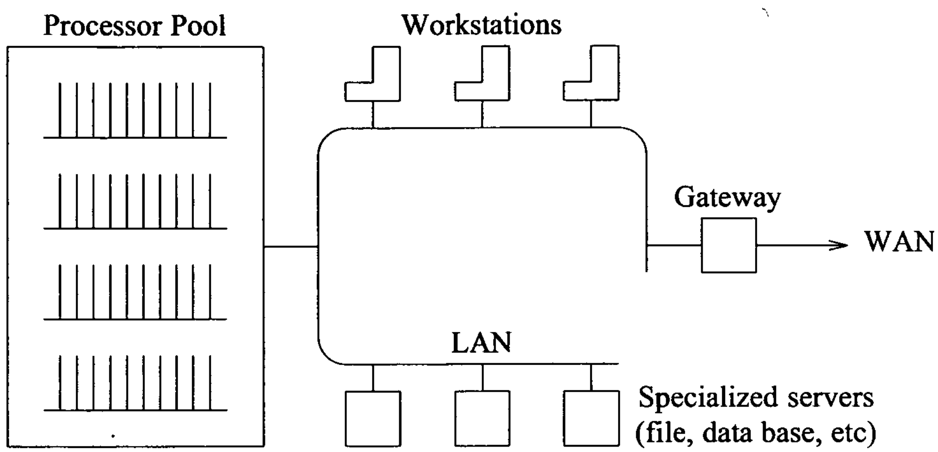


Fig. 1. The four components of the Amoeba architecture.

Amoeba has a *processor pool* for providing most of the computing power. It typically consists of a large number of single-board computers, each with several megabytes of private memory and a network interface. The VU has 48 such machines, for example. A pile of diskless, terminalless workstations can also be used as a processor pool.

When a user has an application to run, e.g., a *make* of a program consisting of dozens of source files, a number of processors can be allocated to run many compilations in parallel. When the user is finished, the processors are returned to the pool so they can be used for other work. Although the pool processors are all multiprogrammed, the best performance is obtained by giving each process its own processor, until the supply runs out.

It is the processor pool that allows us to build a system in which the number of processors exceeds the number of users by an order of magnitude or more, something quite impossible in the personal workstation model of the 1980s. The software has been designed to treat the number of processors dynamically, so new ones can be added as the user population grows. Furthermore, when a few processors crash, some jobs may have to be restarted, and the computing capacity is temporarily lowered, but essentially the system continues normally, providing a degree of fault tolerance.

The third system component consists of the *specialized servers*. These are machines that run dedicated processes that have unusual resource demands. For example, it is best to run file servers on machines that have disks, in order to optimize performance.

Finally, there are gateways to other Amoeba systems far away. In the context of a project sponsored by the European Community, we built a distributed Amoeba system that spanned several countries. The role of the gateway is to protect the local machines from the idiosyncracies of the protocols that must be used over the wide area links.

Why did we choose this architecture as opposed to the traditional workstation model?

Primarily because we believe that the workstation model will become inappropriate in the 1990s, as it becomes possible to give each user 10 or 100 processors. By centralizing the computing power, we allow incremental growth, fault tolerance, and the ability for a single large job to temporarily obtain a large amount of computing power. Current systems have file systems, so why not let them have computer servers as well?

2.2. The Amoeba Software Architecture

Amoeba is an *object-based* system using clients and servers. Client processes use remote procedure calls to send requests to server processes for carrying out operations on objects. Each object is both identified and protected by a *capability*, as shown in Fig. 2. Capabilities have the set of operations that the holder may carry out on the object coded into them and they contain enough redundancy and cryptographic protection to make it infeasible to guess an object's capability. Thus, keeping capabilities secret by embedding them in a huge address space is the key to protection in Amoeba. Due to the cryptographic protection, capabilities can be managed outside the kernel, by user processes themselves.

48	24	8	48	Bits
Service port	Object number	Rights field	Check field	

Fig. 2. Structure of a capability. The service port identifies the service that manages the object. The object number specifies which object (e.g., which file). The rights tell which operations are permitted. The check field provides cryptographic protection to keep users from tampering with the other fields.

Objects are implemented by server processes that manage them. Capabilities have the identity of the object's server encoded into them (the Service Port) so that, given a capability, the system can easily find a server process that manages the corresponding object. The RPC system guarantees that requests and replies are delivered at most once and only to authorized processes. Communication and protection are discussed in Section 3.

Although, at the system level, objects are identified by their (binary) capabilities, at the level where most people program and do their work, objects are named using a symbolic hierarchical naming scheme. The mapping is carried out by the *Directory Service* which maintains a mapping of ASCII path names onto capabilities. The Directory Service has mechanisms for performing atomic operations on arbitrary collections of name-to-capability mappings. The Directory Service is described in Section 4.

Amoeba has already gone through several generations of file systems. Currently, one file server is used practically to exclusion of all others. The Bullet Service, which got its name from being faster than a speeding Bullet, is a simple file server that stores immutable files as contiguous byte strings both on disk and in its cache. It is also described later, in Section 4.

The Amoeba kernel manages memory segments, supports processes containing multiple threads and handles interprocess communication. The process-management facilities allow remote process creation, debugging, checkpointing, and migration, all using a few simple mechanisms explained in Section 5.

All other services, (such as the directory service) are provided by user-level processes, in contrast to, say, Unix, which has a large monolithic kernel that provides these services. By

putting as much as possible in user space, we have achieved a flexible system, and have done this without sacrificing performance.

In the Amoeba design, concessions to existing operating systems and software were carefully avoided. Since it is useful to be able to run existing software on Amoeba, a Unix emulation service, called *Ajax* has been developed. It is discussed in Section 6.

3. COMMUNICATION IN AMOEBA

Amoeba's conceptual model is that of a client thread (light-weight process) performing operations on objects. For example, on a file object, a common operation is reading data from it. Operations are implemented by making *remote procedure calls* [Birrell and Nelson, 1984]. A client sends a *request* message to the *service* that manages the object. A server thread accepts the message, carries out the request, and sends a *reply* message back to the client. For reasons of performance and fault tolerance, frequently multiple server processes jointly manage a collection of objects of the same type to provide a *service*.

3.1. Remote Procedure Calls

The kernel provides three basic system calls to user processes:

- do_operation
- get_request
- send_reply

The first is used by clients to get work done. It consists of sending a message to a server and then blocking until a reply comes back. The second is used by servers to announce their willingness to accept messages addressed to a specific port. The third is also used by servers, to send replies back. All communication in Amoeba is of the form: a client sends a request to a server, the server accepts the request, does the work, and sends back the reply.

Although systems programmers would no doubt be content to live with only these three system calls, for most application programmers they are far too primitive. For this reason a much more user-oriented interface has been built on top of this mechanism, to allow users to think directly in terms of objects and operations on these objects.

Corresponding to each type of object is a *class*. Classes can be composed hierarchically; that is, a class may contain the operations from one or more underlying classes. This *multiple inheritance* mechanism allows many services to inherit the same interfaces for simple object manipulations, such as for changing the protection properties on an object, or deleting an object. It also allows all servers manipulating objects with file-like properties to inherit the same interface for low-level file I/O (read, write, append). The mechanism resembles the file-like properties of Unix pipe and device I/O: the Unix *read* and *write* system calls can be used on files, terminals, pipes, tapes and other I/O devices. But for more detailed manipulation, specialized calls are available (*ioctl*, *popen*, etc.).

Interfaces for object manipulation are specified in a notation, called the Amoeba Interface Language (AIL) [Van Rossum, 1989], which resembles the notation for procedure headers in C with some extra syntax added. This allows automatic generation of client and server stubs. The Amoeba class for standard manipulations on file-like objects, for instance, could be specified as follows:

```
class basic_io [1000..1199] {

    const BIO_SIZE = 30000;

    bio_read(*, in unsigned offset, in out unsigned bytes,
        out char buffer[bytes:bytes]);

    bio_write(*, in unsigned offset, in out unsigned bytes,
        in char buffer[bytes:BIO_SIZE]);
};
```

The names of the operations, *bio_read* and *bio_write*, must be globally unique and conventionally start with an abbreviation of the name of the class they belong to. The first parameter is always a capability of the object to which the operation refers. It is indicated by an asterisk. The other parameters are labelled *in*, *out*, or *in out* to indicate whether they are input or output parameters to the operation, or both. Specifying this allows the stub compiler to generate code to transport parameters in only one direction.

The number of elements in an array parameter can be specified by [*n*: *m*], where *n* is the actual number of elements in the array and *m* is the maximum number. In an *out* array parameter, such as *buffer* in *bio_read*, the maximum size is provided by the caller. In *bio_read*, it is the value of the *in* parameter *bytes*. The actual size of an *out* array parameter is given by the callee and must be less than the maximum. In *bio_read* it is the value of the *out* parameter *bytes* — the actual number of bytes read. On an *in* array parameter, the maximum size is set by the interface designer and must be a constant, while the actual size is given by the caller. In *bio_write*, it is the *in* value of *bytes*.

This AIL specification tells the stub compiler that the operation codes for *basic_io* must be allocated in the range 1000 to 1199. A clash of the operation codes for two different classes only matters if these classes are both inherited by another, bringing them together in one interface. Currently, every group of people designing interfaces has a different range from which to allocate operation codes. At a later stage, we plan to do the allocation of operation codes automatically.

The AIL stub compiler can generate client and server stubs routines for a number of programming languages and machine architectures. For each parameter type, *marshalling code* is compiled into the stubs which converts data types of the language to data types and internal representations of AIL. Currently, AIL handles only fairly simple data types (boolean, integer, floating point, character, string) and records or arrays of them. AIL, however, can easily be extended with more data types when the need arises.

3.2. RPC Transport

The AIL compiler generates code to marshal or unmarshal the parameters of remote procedure calls into and out of message buffers and then call the Amoeba's transport mechanism for the delivery of request and reply messages. Messages consist of two parts, a *header* and a *buffer*. The header has a fixed format and contains addressing information (including the capability of the object that the RPC refers to), an operation code which selects the function to be called on the object, and some space for additional parameters. The buffer can contain data. A file read or write call, for instance, uses the message header for the operation code plus the length and offset parameters, and the buffer for the file data. With this set-up, marshalling the file data (a character array) takes zero time, because the data can be transmitted directly from and to the arguments specified by the program.

The transport interface for the server consists of the calls *get_request* and *send_reply* as described above. They are generally part of a loop that accepts messages, does the work, and sends back replies, like this fragment in C:

```
/* Code for allocating a request buffer */
do {
      get_request(&port, &reqheader, &reqbuffer, reqbuflen);
      /* Code for unmarshalling the request parameters */
      /* Call the implementation routine */
      /* Code for marshalling the reply parameters */
      send_reply(&repheader, &repbuffer, repbuflen);
} while (1);
```

Get_request blocks until a request comes in. *Put_reply* blocks until the header and buffer parameters can be reused. A client sends a request and waits for a reply by calling

```
do_operation(reqheader, reqbuffer, reqbuflen,
      repheader, repbuffer, repbuflen);
```

All of this code is generated automatically by the AIL compiler from the object and operation descriptions given to it.

3.3. Locating Objects

Before a request for an operation on an object can be delivered to a server thread that manages the object, the location of such a thread must be found. All capabilities contain a Service Port field, which identifies the service that manages the object the capability refers to. When a server thread makes a *get_request* call, it provides its service port to the kernel, which records it in an internal table. When a client thread calls *do_operation*, it is the kernel's job to find a server thread with an outstanding *get_request* that matches the port in the capability provided by the client.

We call the process of finding the address of such a server thread *locating*. It works as follows. When a *do_operation* call comes into a kernel, a check is made to see if the port in question is already known. If not, the kernel broadcasts a special *locate* packet onto the network asking if anyone out there has an outstanding *get_request* for the port in question. If one or more kernels have servers with outstanding *get_request*s they respond by sending their network addresses. The kernel doing the broadcasting records the (port, network address) pair in a cache for future use. Only if a server dies or migrates will another broadcast be needed.

When Amoeba is run over a wide area network, with huge numbers of machines, a slightly different scheme is used. Each server wishing to export its service sends a special message to all the domains in which it wants its service known. (A *domain* could be a company, campus, city, country or something else.) In each such domain, a dummy process, called a *server agent* is created. This process does a *get_request* using the server's port and then lies dormant until a request comes in, at which time it forwards the message to the server for processing. Note that a port is just a randomly chosen 48-bit number. It in no way identifies a particular domain, network, or machine.

3.4. Secure Communication

Client requests, addressed using an object's capability are delivered to one of the servers with outstanding *get_request* calls on the capability's port. Ports consist of large, 48-bit numbers which are known only to the server processes that comprise the service, and to the server's clients. For a public service, such as the file system, the port will generally be made known to all users. The ports used by an ordinary user process will, in general, be kept secret. Knowledge of a port is taken by the system as *prima facie* evidence that the sender has a right to communicate with the service. Of course, the service is not required to carry out work for clients just because they know the port, for example, the file server will refuse to read or write files for clients lacking appropriate file capabilities. Thus two levels of protection are used in Amoeba: ports for protecting access to servers, and capabilities for protecting access to individual objects.

Although the port mechanism provides a convenient way to provide partial authentication of clients ('if you know the port, you may at least talk to the service'), it does not deal with the authentication of servers. How does one ensure that malicious users do not make *get_request* calls on the file server's port, and try to impersonate the file server to the rest of the system?

One approach is to have all ports manipulated by kernels that are presumed to be trustworthy and are supposed to know who may listen on which port. We have rejected this strategy because on some machines, such as personal computers, users may be able to tamper with the operating system kernel, and also because we believe that by making the kernel as small as possible, we can enhance the reliability of the system as a whole. Instead, we have chosen a different solution that can be implemented in either hardware or software.

In the hardware solution, we need to place a small interface box, which we call an F-box (Function-box) between each processor module and the network. The most logical place to put it is on the VLSI chip that is used to interface to the network. Alternatively, it can be put on a small printed circuit board inside the wall socket through which personal computers attach to the network. In those cases where the processors have user mode and kernel mode and the operating systems can be trusted, it could be put into the operating system. This is the solution in the current Amoeba implementation.

In the software solution, we build the F-box out of cryptographic algorithms, giving the same functional effect as the hardware F-box. In any event, we assume that, somehow or other, all messages entering and leaving every processor undergo a simple transformation that users cannot bypass.

The transformation works like this. Each port is really a pair of ports, P, and G, related by: $P = F(G)$, where F is a (publicly-known) one-way function [Wilkes, 1968] performed by the F-box. The one-way function has the property that given G it is a straightforward computation to find P, but that given P, finding G is not feasible.

Using the one-way F-box, the server authentication can be handled in a simple way, as illustrated in Fig. 3. Each server chooses a get-port, G, and computes the corresponding put-port, P. The get-port is kept secret; the put-port is distributed to potential clients or, in the case of public servers, is published. When the server is ready to accept client requests, it does a *get_request*(G, $\cdots$). The F-box then computes $P = F(G)$ and waits for messages containing P to arrive. When one arrives, it is given to the server process. To send a message to the server, the client merely does *do_operation*(P, $\cdots$), which sends a message containing P in a header field to the server. The F-box on the sender's side does not perform any transformation on the P field of the outgoing message.

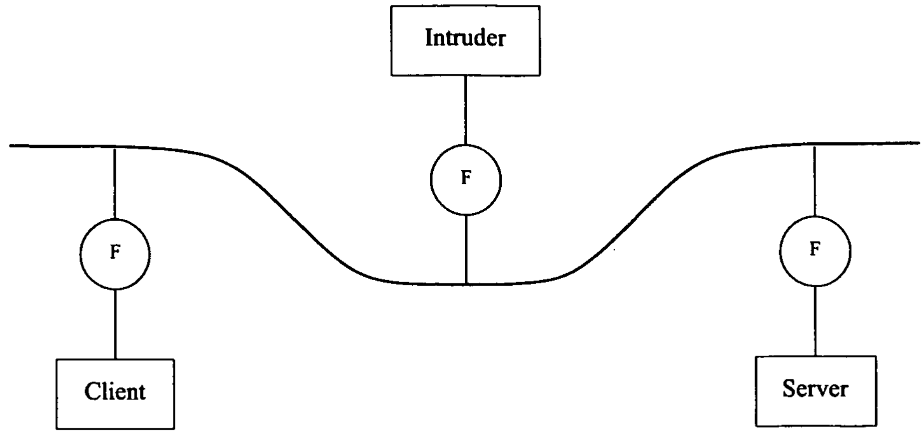


Fig. 3. Clients, servers, intruders, and F-boxes.

Now let us consider the system from an intruder's point of view. To impersonate a server, the intruder must do *get_request*(G, $\cdots$). However, G is a well-kept secret, and is never transmitted on the network. Since we have assumed that G cannot be deduced from P (the one-way property of F) and that the F-box cannot be circumvented, the intruder cannot intercept messages not intended for him. An intruder doing *get_request*(P, $\cdots$) will simply cause his F-box to listen to the (useless) port F(P). Replies from the server to the client are protected the same way, only with the client picking a get-port for the reply, say, G', and including $P' = F(G')$ in the request message.

The presence of the F-box makes it easy to implement digital signatures for further authentication still, if that is desired. To do so, each client chooses a random signature, S, and publishes $F(S)$. The F-box must be designed to work as follows. Each message presented to the F-box for transmission contains three special header fields: destination (P), reply (G'), and signature (S). The F-box applies the one-way function to the second and third of these, transmitting the three ports as: P, $F(G')$, and $F(S)$, respectively. The first is used by the receiver's F-box to admit only those messages for which the corresponding *get* has been done, the second is used as the put-port for the reply, and the third can be used to authenticate the sender, since only the true owner of the signature will know what number to put in the third field to insure that the publicly-known $F(S)$ comes out.

It is important to note that the F-box arrangement merely provides a simple *mechanism* for implementing security and protection, but gives operating system designers considerable latitude for choosing various *policies*. The mechanism is sufficiently flexible and general that it should be possible to put it into hardware without precluding many as-yet-unthought-of operating systems to be designed in the future.

3.5. Performance of Amoeba RPC

To measure the speed of the Amoeba RPC, we ran some timing tests. For example, we booted the Amoeba kernel on two 16.7 MHz Motorola 68020s and created a user process on each and let them communicate over a 10 Mbps Ethernet. For a message consisting of just a header (no data), the complete RPC took 1.4 msec. With 8K of data it took 13.1 msec, and with 30K it took 44.0 msec. The latter corresponds to a throughput of 5.4 megabits/sec, which is half the theoretical capacity of the Ethernet, and much higher than most other systems achieve. Five client-server pairs together can achieve a total throughput of 8.4 megabits per second, not counting Ethernet and Amoeba packet headers.

Fig. 4 shows the speeds and throughput of local communication (communication between processes on the same machine) and remote communication (processes on different machines communicating over the Ethernet). Remote operations were carried out with requests containing 4 bytes, 8 kilobytes and 30 kilobytes, and empty replies. Three RPC implementations were measured: RPC on native Amoeba, the same Amoeba protocol used from a driver under SUN Unix, and SUN's own RPC.

	Delay (msec)		
	case 1 *(4 bytes)*	case 2 *(8 Kb)*	case 3 *(30 Kb)*
bare Amoeba local	0.8	2.5	7.1
bare Amoeba remote	1.4	13.1	44.0
UNIX driver local	4.5	10.0	32.0
UNIX driver remote	7.0	36.4	134.0
SUN RPC local	10.4	23.6	imposs.
SUN RPC remote	12.2	40.6	imposs.

(a)

	Bandwidth (Kbytes/sec)		
	case 1 *(4 bytes)*	case 2 *(8 Kb)*	case 3 *(30 Kb)*
bare Amoeba local	5.0	3,277	4,255
bare Amoeba remote	2.9	625	677
UNIX driver local	0.9	819	938
UNIX driver remote	0.6	225	224
SUN RPC local	0.4	347	imposs.
SUN RPC remote	0.3	202	imposs.

(b)

Fig. 4. The delay in msec (a) and bandwidth in Kbytes/sec (b) for RPC between user processes in three common cases for three different systems. Local RPCs are RPCs where the client and server are running on the same processor. The Unix driver implements Amoeba RPCs under SUN Unix

Why did we use objects, capabilities, and RPC as the base for the design? Objects are a natural way to program. By encapsulating information, users are forced to pay attention to precise interfaces and irrelevant information is hidden from them. Capabilities are a clean and elegant way to name and protect objects. By using an encryption scheme for protecting them, we moved the capability management out of the kernel. RPC is an obvious way to implement the request/reply nature of performing operations on objects.

4. THE AMOEBA FILE SYSTEM

Capabilities form the low-level naming mechanism of Amoeba, but they are very impractical for use by human beings. Therefore an extra level of mapping is provided from symbolic hierarchical path names to capabilities. On Amoeba, a typical user has access to literally thousands of capabilities — of the user's own private objects, but also capabilities of public objects, such as the executables of commands, pool processors, data bases, public files, and so on.

It is perhaps feasible for a user to store his own private capabilities somewhere, but it is quite impossible for a system manager, or a project co-ordinator to hand out capabilities explicitly to every user who may access a shared public object. Public places are needed where users can find capabilities of shared objects, so that when a new object is made sharable, or when a sharable object changes, its capability need be put in only one place so everyone can find it easily.

4.1. The Hierarchical Directory Structure

Hierarchical directory structures are ideal for implementing partially shared name spaces. Objects that are shared between members of a project team can be stored in a directory that only team members have access to. By implementing directories as ordinary objects with a capability that is needed to use them, members of a group can be given access by giving them

the capability of the directory, while others can be withheld access by not giving them the capability. A capability for a directory is thus a capability for many other capabilities.

To a first approximation, a directory is a set of (name, capability) pairs. The basic operations on directory objects are:

- lookup
- enter
- delete

The first one looks up an object name in a directory and returns its capability. The other two enter and delete objects from directories. Since directories themselves are objects, a directory may contain capabilities for other directories, thus potentially allowing users to build an arbitrary graph structure.

Complex sharing can be achieved by making directories more sophisticated than we have just described. In reality, a directory is an $(n+1)$-column table with ASCII names in column 0 and capabilities in columns 1 through n. A capability for a directory is really a capability for a specific column of a directory. Thus, for example, a user could arrange his directories with one column for himself, a second column for members of his group, and a third column for everyone else. This scheme can provide the same protection rules as Unix, but obviously many other schemes are also possible.

The Directory Service can be set up so that whenever a new object is entered in a directory, the Directory Service first asks the service managing the object to make n replicas, potentially physically distributed for reliability. All the capabilities are then entered into the directory.

4.2. The Bullet Service

The Bullet Service is a highly unusual file server. Each of the Bullet Servers support only three principal operations:

- read_file
- create_file
- delete_file

When a file is created, the user normally provides all the data at once, creating the file and getting back a capability for it. In most circumstances the user will immediately give the file a name and ask the Directory Service to enter the (name, capability) pair in some directory.

All files are *immutable*, that is, once created they cannot be changed. Notice that there is no *write* operation supported. Since files cannot change, the Directory Service can replicate them at its leisure for redundancy without fear that a file may change in the meanwhile.

Since the final file size is known when a file is created, files can, and are, stored contiguously, both on the disk and in Bullet Servers' caches, as illustrated in Fig. 5. The administrative information for a file is then reduced to its origin and size plus some ownership data. The complete administrative table is loaded into the Bullet Server's memory when it is booted. When a *read* operation is done, the object number in the capability is used as an index into this table, and the file is read into the cache in a single (possibly multitrack) disk operation.

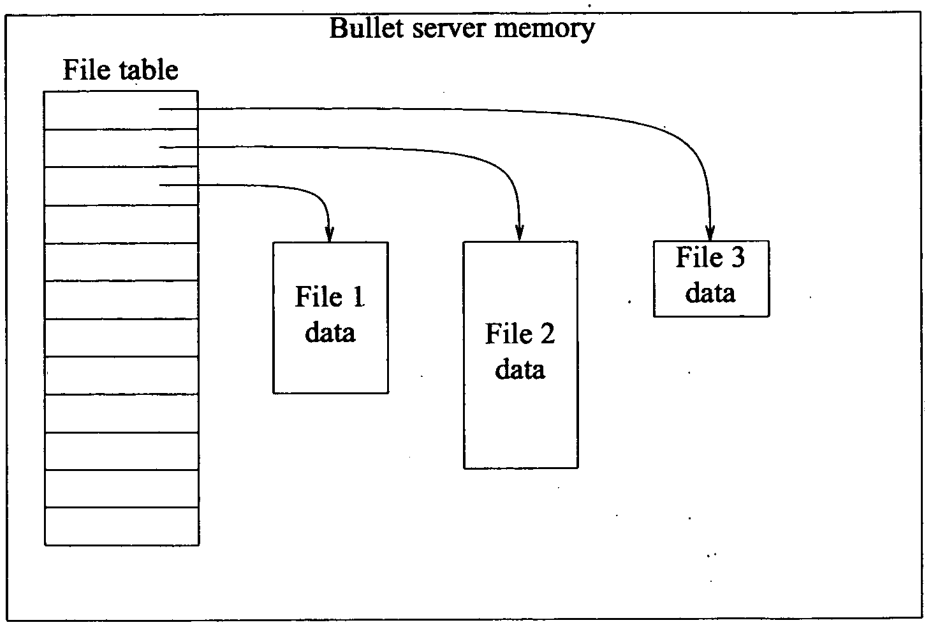


Fig. 5. Bullet Server file representation.

The Bullet file service can deliver large files from its cache, or consume large files into its cache at maximum RPC speeds, that is, at 677 kilobytes per second. Reading a 4 kilobyte file from a Bullet Server's cache by a remote client (over the Ethernet) takes 7 msec; a 1 megabyte file takes 1.6 sec. More detailed performance numbers and comparisons with other systems can be found in [Van Renesse et al., 1989].

Although the Bullet Service wastes some space due to fragmentation, its performance easily compensates for having to buy an 800M disk to store, say, 500M worth of data.

4.3. Atomicity

Ideally, names always refer to consistent objects and sets of names always refer to mutually consistent sets of objects. In practice, this is seldom the case and it is, in fact, not always necessary or desirable. But there are many cases where it is necessary to have consistency.

Atomic actions form a useful tool for achieving consistent updates to sets of objects. Protocols for atomic updates are well understood and it is possible to provide a toolkit which allows independently implemented services to collaborate in atomic updates of multiple objects managed by several services.

In Amoeba, a different approach to atomic updates has been chosen. The Directory Service takes care of atomic updates by allowing the mapping of arbitrary sets of names onto arbitrary sets of capabilities to be changed atomically. The objects referred to by these capabilities must be immutable, either because the services that manage them refuse to change them (e.g., the Bullet Service) or because the users refrain from changing them.

The atomic transactions provided by the Directory Service are not particularly useful for dedicated transaction-processing applications (e.g., banking, or airline-reservation systems), but they are useful in preventing the glitches that sometimes result from users using an application just when a new version is installed, or two people simultaneously updating a file resulting in one lost update.

4.4. Reliability and Security

The Directory Service plays a crucial role in the system. Nearly every application depends on it for finding the capabilities it needs. If the Directory Service stops, everything else will come to a halt as well. Thus the Directory Service must never stop.

The Directory Service replicates all its internal tables on multiple disks so that no single-site failure will bring it down. The techniques used to achieve this are essentially the same techniques used in fault-tolerant data base systems.

The Directory Service is not only relied on to be up and running; it is also trusted to work correctly and never divulge a capability to an entity that is not entitled to see it. Security is an important aspect of the reliability of the directory service.

Even a perfect design of the Directory Service may lead to unauthorized users catching glimpses of the data stored in it. Hardware diagnostic software, for example, has access to the Directory Server's disk storage. Bugs in the operating system kernel might allow users to read portions of the disk.

Directories may be encrypted in order to prevent bugs in the directory server, in the operating system or other idiosyncrasies from laying bare the confidential information stored in them. The encryption key may be exclusive-or'ed with a random number and the result may be stored alongside the directory, while the random number is put in the directory's capability. After giving the capability to the owner, the Directory Service itself can forget the random number. It only needs it when the directory has to be decrypted in order to carry out operations on it, and will always receive the random number in the capability which comes with every client's request.

Why did we design such an unconventional file system? Partly to achieve great speed and partly for simplicity in design and implementation. The use of immutable files (and some other objects) makes it possible to centralize the replication mechanism in one place — the Directory Service. Immutable files are also easy to cache (because a cached immutable file can never become stale), an important issue when Amoeba is run over wide area networks.

5. PROCESS MANAGEMENT

Amoeba processes can have multiple threads of control. A process consists of a segmented virtual address space and one or more threads. Processes can be remotely created, destroyed, checkpointed, migrated and debugged.

On a uniprocessor, threads run in quasi-parallel; on a shared-memory multiprocessor, as many threads can run simultaneously as there are processors. Processes cannot be split up over more than one machine.

Processes have explicit control over their address space. They can add new segments to their address space by *mapping them in* and remove segments by *mapping them out*. Besides virtual address and length, a capability can be specified in a map operation. This capability must belong to a file-like object which is read by the kernel to initialize the new segment. This allows processes to do mapped-file I/O.

When a segment is mapped out, it remains in memory, although no longer as part of any process' address space. The unmap operation returns a capability for the segment which can then be read and written like a file. One process can thus map a segment out and pass the capability to another process; the other process can then map the segment in again. If the processes are on different machines, the contents of the segment are copied (by one kernel doing read operations and the other kernel servicing them); on the same machine, the kernel can use shortcuts for the same effect.

A process is created by sending a *process descriptor* to a kernel in an *execute process* request. A process descriptor consists of four parts as shown in Fig. 6. The host descriptor describes on what machine the process may run, e.g., its instruction set, extended instruction sets (when required), memory needs, etc., but also it can specify a class of machines, a group of machines or a particular machine. A kernel that does not match the host descriptor will refuse to execute the process.

The capabilities are next. One is the capability of the process which every client that manipulates the process needs. Another is the capability of a *handler*, a service that deals with process exit, exceptions, signals and other anomalies of the process.

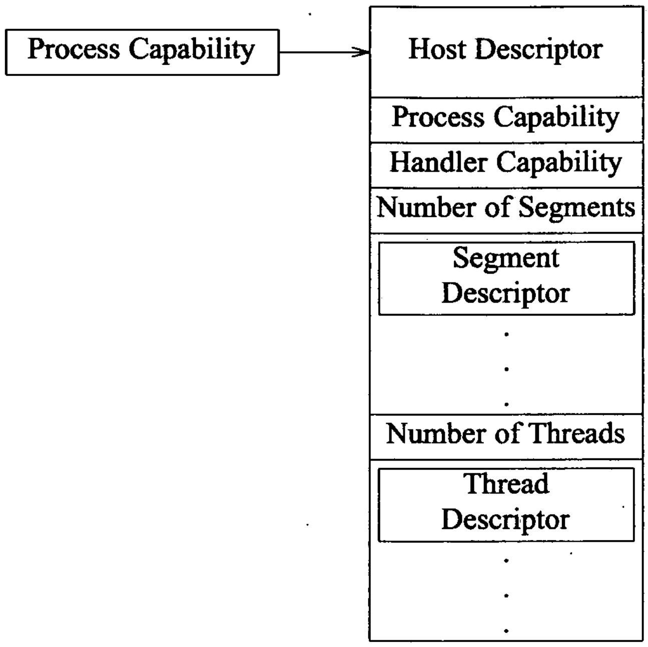


Fig. 6. Layout of a process descriptor.

The memory map has an entry for each segment in the address space of the process to be. An entry gives virtual address, segment length, how the segment should be mapped (read only, read/write, execute only, etc.), and the capability of a file or segment from which the new segment should be initialized.

The thread map describes the initial state of each of the threads in the new process, processor status word, program counter, stack pointer, stack base, register values, and system call state. This rather elaborate notion of thread state allows the use of process descriptors not only for the representation of executable files, but also for processes being migrated debugged or being checkpointed.

In most operating systems, system call state is large and complicated to represent outside an operating system kernel. In Amoeba, fortunately, there are very few system calls that can block in the kernel. The most complicated ones are those for communication: *do_operation* and *get_request*.

Processes can be in two states, *running*, or *stunned*. In the stunned state, a process exists, but does not execute instructions. A process being debugged is in the stunned state, for example. The low-level communication protocols in the operating system kernel respond

with 'this-process-is-stunned' messages to attempts to communicate with the process. The sending kernel will keep trying to communicate until the process is running again or until it is killed. Thus, communication with a process being interactively debugged continues to work.

A running process can be stunned by a stun request directed to it from the outside world (this requires the stunner to have the capability of the process as evidence of ownership), or by an uncaught exception. When the process becomes stunned, the kernel sends its state in a process descriptor to a *handler* whose identity is a capability which is part of the process' state. After examining the process descriptor, and possibly modifying it or the stunned process' memory, the handler can reply either with a *resume* or *kill* command.

Debugging processes is done with this mechanism. The debugger takes the role of the handler. Migration is also done through stunning. First, the candidate process is stunned; then, the handler gives the process descriptor to the new host. The new host fetches memory contents from the old host in a series of file read requests, starts the process and returns the capability of the new process to the handler. Finally, the handler returns a *kill* reply to the old host. Processes communicating with a process being migrated will receive 'process-is-stunned' replies to their attempts until the process on the old host is killed. They will then get a 'process-not-here' reaction. After locating the process again, communication will resume with the process on the new host.

The mechanism allows command interpreters to cache process descriptors of the programs they start and it allows kernels to cache code segments of the processes they run. Combined, these caching techniques make process start-up times very short.

Our process management mechanisms are unusual, but they are intended for an unusual environment: one where remote execution is the normal case and local execution is the exception. The boundary conditions for our design were the creation of a few simple mechanisms that allowed us to do process execution, migration, debugging and checkpointing in such a way that a very efficient implementation is possible.

6. UNIX EMULATION

Amoeba is a new operating system with a system interface that is quite different from that of the popular operating systems of today. Since we had no intention of writing hundreds of utility programs for Amoeba from scratch, it was quickly decided to write a Unix emulation package to allow most Unix utilities to work on Amoeba, sometimes with small changes. Binary compatibility was considered as a possibility, but was rejected for an initial emulation package on grounds that it is more complicated and less useful (first, one has to choose a very particular version of Unix; second, one usually has binaries for only one machine architecture, while sources can be compiled for any machine archtecture; and, third, binary emulation is bound to be slow).

The emulation facility started out as a library of Unix routines that have the standard Unix interface and semantics, but do their work by calling the Bullet Service, the Directory Service and the Amoeba process management facilities. The system calls implemented initially were those for file I/O (*open, close, dup, read, write, lseek*) and a few of the *ioctl* calls for ttys. These were very easy to implement under Amoeba (about two week's work) and were enough to get a surprising number of Unix utilities to run.

Subsequently, a *Session* server was developed to allocate Unix PIDs, PPIDs, and assist in the handling of system calls involving them (*fork, exec, signal, kill*). The Session Server is also used for dealing with Unix *pipes*. With the help of the Session Server many other Unix utilities are now usable on Amoeba.

Currently, about 100 utilities have been made to run on Amoeba without any changes to

the source code. We have not attempted to port some of the more esoteric Unix programs. Work is in progress to make our Unix interface compatible with the emerging standards (e.g., IEEE POSIX).

The X window system has been ported to Amoeba and supports the use of both TCP/IP and Amoeba RPC, so that an X client on Amoeba can converse with an X server on Amoeba and vice versa.

We have found that the availability of the Unix utilities has made the transition to Amoeba much easier. Slowly, however, many of the Unix utilities will be replaced by utilities that are better adapted to the Amoeba distributed environment. Our new parallel *make* is an obvious example.

Why did we emulate Unix in a library instead of making the system binary compatible? Because any system that is binary compatible with Unix cannot be much of a step forward beyond the ideas of the early 1970s. We wanted to design a new system from the ground up for the 1990s. If the Unix designers had constrained themselves to be binary compatible with the then-popular RT-11 operating system, it would not be where it is now.

7. CONCLUSIONS

We are pleased with most of the design decisions of the Amoeba project. The decision, especially, to design a distributed operating system without attempting to restrict ourselves to existing operating systems or operating system interfaces has been a good one. Unix is an excellent operating system, but it is not a distributed one and was not designed as such. We do not believe we would have made such a balanced design had we decided to build a distributed system with a Unix interface.

In spite of our design-independence from Unix, we found it remarkably easy to port all the Unix software we wanted to use to Amoeba. The programs that are hard to port are mostly those we have no need for in Amoeba (programs for network access and for system maintenance and management, for example).

The use of objects and capabilities has also given us some very important advantages. When a service is being designed, the protection of its objects usually does not require any though; the use of capabilities automatically provides enough of a protection mechanism. It also gave us a very uniform and decentralized object-naming and -access mechanism.

The decision not to build on top of an existing operating system, but to build directly on the hardware has been absolutely essential to the success of Amoeba. One of the primary goals of the project was to design and build a high-performance system and this can hardly be done on top of another system. As far as we can tell, only systems with custom-built hardware or special microcode can outperform Amoeba's RPC and file system on comparable hardware.

The Amoeba kernel is small and simple. It implements only a few operations for process management, and interprocess communication, but they are versatile and easy to use. The performance of its interprocess communication has already been mentioned. The kernel is easy to port between hardware platforms. It now runs on VAX and Motorola 68020 and 68030 processors, and is currently being ported to the Intel 80386. Amoeba is now available. For information about how to obtain a copy, please contact the authors.

8. REFERENCES

M. Accetta, R. Baron, W. Bolosky, D. Golub, R. Rashid, A. Tevanian, and M. Young [1986].
Mach: A New Kernel Foundation for UNIX Development.
Proceedings of the Summer Usenix Conference, Atlanta, GA, July 1986.

A. D. Birrell and B. J. Nelson [1984].
Implementing Remote Procedure Calls.
ACM Transactions on Computer Systems 2 (1) : 39–59, February 1984.

D. R. Cheriton [1988].
The V Distributed System.
Communications of the ACM 31 : 314–333, March 1988.

S. J. Mullender and A. S. Tanenbaum [1986].
The Design of a Capability-Based Distributed Operating System.
The Computer Journal 29 (4) : 289–300, 1986.

R. M. Needham and A. J. Herbert [1982].
The Cambridge Distributed Computing System.
Addison Wesley, Reading, MA, 1982.

M. Rozier, V. Abrossimov, F. Armand, I. Boule, M. Gien, M. Guillemont, F. Hermann, C. Kaiser, S. Langlois, P. Léonard, and W. Neuhauser [1988].
CHORUS Distributed Operating Systems.
Report CS/Technical Report-88-7.6, Chorus Systèmes, Paris, Nov. 1988.

R. van Renesse, J. M. van Staveren, and A. S. Tanenbaum [1989].
Performance of the Amoeba Distributed Operating System.
Software–Practice and Experience 19 : 223–234, March 1989.

G. van Rossum [1989].
AIL – A Class-Oriented Stub Generator for Amoeba.
In W. Zimmer, editor, *Proceedings of the Workshop on Experience with Distributed Systems.* Springer Verlag, 1989.
To appear.

B. Walker, G. Popek, R. English, C. Kline, and G. Thiel [1983].
The LOCUS Distributed Operating System.
Proceedings Ninth Symposium on Operating System Principles : 49–70, 1983.

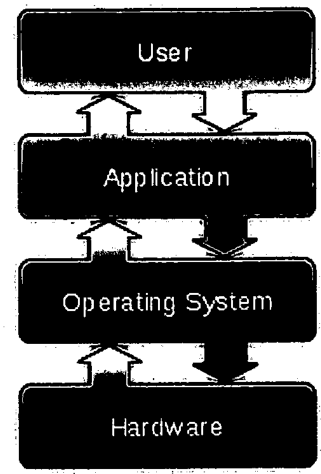
M. V. Wilkes [1968].
Time-Sharing Computer Systems.
American Elsevier, New York, 1968.

Operating system

From Wikipedia, the free encyclopedia

An **operating system (OS)** is an interface between hardware and user which is responsible for the management and coordination of activities and the sharing of the resources of the computer that acts as a host for computing applications run on the machine. As a host, one of the purposes of an operating system is to handle the details of the operation of the hardware. This relieves application programs from having to manage these details and makes it easier to write applications. Almost all computers (including handheld computers, desktop computers, supercomputers, video game consoles) as well as some robots, domestic appliances (dishwashers, washing machines), and portable media players use an operating system of some type.[1] Some of the oldest models may, however, use an embedded operating system that may be contained on a data storage device.

Operating systems



Common features

- Process management
- Interrupts
- Memory management
- Virtual file system
- Device drivers
- Networking
- Security
- Graphical user interfaces

Operating systems offer a number of services to application programs and users. Applications access these services through application programming interfaces (APIs) or system calls. By invoking these interfaces, the application can request a service from the operating system, pass parameters, and receive the results of the operation. Users may also interact with the operating system with some kind of software user interface (SUI) like typing commands by using command line interface (CLI) or using a graphical user interface (GUI, commonly pronounced “gooey”). For hand-held and desktop computers, the user interface is generally considered part of the operating system. On large multi-user systems like Unix and Unix-like systems, the user interface is generally implemented as an application program that runs outside the operating system. (Whether the user interface should be included as part of the operating system is a point of contention.)

Common contemporary operating systems include BSD, Darwin (Mac OS X), Linux, SunOS (Solaris/OpenSolaris), and Windows NT (XP/Vista/7). While servers generally run Unix or some Unix-like operating system, embedded system markets are split amongst several operating systems,[2][3] although the Microsoft Windows line of operating systems has almost 90% of the client PC market.

Contents

History

Main article: History of operating systems

Mainframes

Through the 1950s, many major features were pioneered in the field of operating systems. The development of the IBM System/360 produced a family of mainframe computers available in widely differing capacities and price points, for which a single operating system OS/360 was planned (rather than developing ad-hoc programs for every individual model). This concept of a single OS spanning an entire product line was crucial for the success of System/360 and, in fact, IBM`s current mainframe operating systems are distant descendants of this original system; applications written for the OS/360 can still be run on modern machines. In the mid-70's, the MVS, the descendant of OS/360 offered the first implementation of using RAM as a transparent cache for data.

OS/360 also pioneered a number of concepts that, in some cases, are still not seen outside of the mainframe arena. For instance, in OS/360, when a program is started, the operating system keeps track of all of the system resources that are used including storage, locks, data files, and so on. When the process is terminated for any reason, all of these resources are re-claimed by the operating system. An alternative CP-67 system started a whole line of operating systems focused on the concept of virtual machines.

Control Data Corporation developed the SCOPE operating system in the 1960s, for batch processing. In cooperation with the University of Minnesota, the KRONOS and later the NOS operating systems were developed during the 1970s, which supported simultaneous batch and timesharing use. Like many commercial timesharing systems, its interface was an extension of the Dartmouth BASIC operating systems, one of the pioneering efforts in timesharing and programming languages. In the late 1970s, Control Data and the University of Illinois developed the PLATO operating system, which used plasma panel displays and long-distance time sharing networks. Plato was remarkably innovative for its time, featuring real-time chat, and multi-user graphical games. Burroughs Corporation introduced the B5000 in 1961 with the MCP, (Master Control Program) operating system. The B5000 was a stack machine designed to exclusively support high-level languages with no machine language or assembler, and indeed the MCP was the first OS to be written exclusively in a high-level language – ESPOL, a dialect of ALGOL. MCP also introduced many other ground-breaking innovations, such as being the first commercial implementation of virtual memory. During development of the AS400, IBM made an approach to Burroughs to licence MCP to run on the AS400 hardware. This proposal was declined by Burroughs management to protect its existing hardware production. MCP is still in use today in the Unisys ClearPath/MCP line of computers.

UNIVAC, the first commercial computer manufacturer, produced a series of EXEC operating systems. Like all early main-frame systems, this was a batch-oriented system that managed magnetic drums, disks, card readers and line printers. In the 1970s, UNIVAC produced the Real-Time Basic (RTB) system to support large-scale time sharing, also patterned after the Dartmouth BASIC system.

General Electric and MIT developed General Electric Comprehensive Operating Supervisor (GECOS), which introduced the concept of ringed security privilege levels. After acquisition by Honeywell it was renamed to General Comprehensive Operating System (GCOS).

Digital Equipment Corporation developed many operating systems for its various computer lines, including TOPS-10 and TOPS-20 time sharing systems for the 36-bit PDP-10 class systems. Prior to the widespread use of UNIX, TOPS-10 was a particularly popular system in universities, and in the early ARPANET community.

In the late 1960s through the late 1970s, several hardware capabilities evolved that allowed similar or ported software to run on more than one system. Early systems had utilized microprogramming to implement features on their systems in order to permit different underlying architecture to appear to be the same as others in a series. In fact most 360's after the 360/40 (except the 360/165 and 360/168) were microprogrammed implementations. But soon other means of achieving application compatibility were proven to be more significant.

The enormous investment in software for these systems made since 1960s caused most of the original computer manufacturers to continue to develop compatible operating systems along with the hardware. The notable supported mainframe operating systems include:

- Burroughs MCP – B5000,1961 to Unisys Clearpath/MCP, present.
- IBM OS/360 – IBM System/360, 1966 to IBM z/OS, present.
- IBM CP-67 – IBM System/360, 1967 to IBM z/VM, present.
- UNIVAC EXEC 8 – UNIVAC 1108, 1967, to OS 2200 Unisys Clearpath Dorado, present.

Microcomputers

The first microcomputers did not have the capacity or need for the elaborate operating systems that had

been developed for mainframes and minis; minimalistic operating systems were developed, often loaded from ROM and known as *Monitors*. One notable early disk-based operating system was CP/M, which was supported on many early microcomputers and was closely imitated in MS-DOS, which became wildly popular as the operating system chosen for the IBM PC (IBM's version of it was called IBM DOS or PC DOS), its successors making Microsoft. In the 80's Apple Computer Inc. (now Apple Inc.) abandoned its popular Apple II series of microcomputers to introduce the Apple Macintosh computer with an innovative Graphical User Interface (GUI) to the Mac OS.

The introduction of the Intel 80386 CPU chip with 32-bit architecture and paging capabilities, provided personal computers with the ability to run multitasking operating systems like those of earlier minicomputers and mainframes. Microsoft responded to this progress by hiring Dave Cutler, who had developed the VMS operating system for Digital Equipment Corporation. He would lead the development of the Windows NT operating system, which continues to serve as the basis for Microsoft's operating systems line. Steve Jobs, a co-founder of Apple Inc., started NeXT Computer Inc., which developed the Unix-like NEXTSTEP operating system. NEXTSTEP would later be acquired by Apple Inc. and used, along with code from FreeBSD as the core of Mac OS X.

Minix, an academic teaching tool which could be run on early PCs, would inspire another reimplementation of Unix, called Linux. Started by computer science student Linus Torvalds with cooperation from volunteers over the Internet, an operating system was developed with the tools from the GNU Project. The Berkeley Software Distribution, known as BSD, is the UNIX derivative distributed by the University of California, Berkeley, starting in the 1970s. Freely distributed and ported to many minicomputers, it eventually also gained a following for use on PCs, mainly as FreeBSD, NetBSD and OpenBSD.

Features

Program execution

Main article: Process (computing)

The operating system acts as an interface between an application and the hardware. The user interacts with the hardware from "the other side". The operating system is a set of services which simplifies development of applications. Executing a program involves the creation of a process by the operating system. The kernel creates a process by assigning memory and other resources, establishing a priority for the process (in multi-tasking systems), loading program code into memory, and executing the program. The program then interacts with the user and/or other devices and performs its intended function.

Interrupts

Main article: interrupt

Interrupts are central to operating systems, since they provide an efficient way for the operating system to interact with and react to its environment. The alternative—having the operating system "watch" the various sources of input for events (polling) that require action—is a poor use of CPU resources. Interrupt-based programming is directly supported by most CPUs. Interrupts provide a computer with a way of automatically running specific code in response to events. Even very basic computers support hardware interrupts, and allow the programmer to specify code which may be run when that event takes place.

When an interrupt is received, the computer's hardware automatically suspends whatever program is currently running, saves its status, and runs computer code previously associated with the interrupt; this is analogous to placing a bookmark in a book in response to a phone call. In modern operating systems, interrupts are handled by the operating system's kernel. Interrupts may come from either the computer's hardware or from the running program.

When a hardware device triggers an interrupt the operating system's kernel decides how to deal with this event, generally by running some processing code. How much code gets run depends on the priority of the interrupt (for example: a person usually responds to a smoke detector alarm before answering the phone). The processing of hardware interrupts is a task that is usually delegated to software called device drivers, which may be either part of the operating system's kernel, part of another program, or both. Device drivers may then relay information to a running program by various means.

A program may also trigger an interrupt to the operating system. If a program wishes to access hardware for example, it may interrupt the operating system's kernel, which causes control to be passed back to the kernel. The kernel will then process the request. If a program wishes additional resources (or wishes to shed resources) such as memory, it will trigger an interrupt to get the kernel's attention.

Protected mode and supervisor mode

Main article: Protected mode

Main article: Supervisor mode

Modern CPUs support something called dual mode operation. CPUs with this capability use two modes: protected mode and supervisor mode, which allow certain CPU functions to be controlled and affected only by the operating system kernel. Here, protected mode does not refer specifically to the 80286 (Intel's x86 16-bit microprocessor) CPU feature, although its protected mode is very similar to it. CPUs might have other modes similar to 80286 protected mode as well, such as the virtual 8086 mode of the 80386 (Intel's x86 32-bit microprocessor or i386).

However, the term is used here more generally in operating system theory to refer to all modes which limit the capabilities of programs running in that mode, providing things like virtual memory addressing and limiting access to hardware in a manner determined by a program running in supervisor mode. Similar modes have existed in supercomputers, minicomputers, and mainframes as they are essential to fully supporting UNIX-like multi-user operating systems.

When a computer first starts up, it is automatically running in supervisor mode. The first few programs to run on the computer, being the BIOS, bootloader and the operating system have unlimited access to hardware - and this is required because, by definition, initializing a protected environment can only be done outside of one. However, when the operating system passes control to another program, it can place the CPU into protected mode.

In protected mode, programs may have access to a more limited set of the CPU's instructions. A user program may leave protected mode only by triggering an interrupt, causing control to be passed back to the kernel. In this way the operating system can maintain exclusive control over things like access to hardware and memory.

The term "protected mode resource" generally refers to one or more CPU registers, which contain information that the running program isn't allowed to alter. Attempts to alter these resources generally causes a switch to supervisor mode, where the operating system can deal with the illegal operation the

program was attempting (for example, by killing the program).

Memory management

Main article: memory management

Among other things, a multiprogramming operating system kernel must be responsible for managing all system memory which is currently in use by programs. This ensures that a program does not interfere with memory already used by another program. Since programs time share, each program must have independent access to memory.

Cooperative memory management, used by many early operating systems assumes that all programs make voluntary use of the kernel's memory manager, and do not exceed their allocated memory. This system of memory management is almost never seen anymore, since programs often contain bugs which can cause them to exceed their allocated memory. If a program fails it may cause memory used by one or more other programs to be affected or overwritten. Malicious programs, or viruses may purposefully alter another program's memory or may affect the operation of the operating system itself. With cooperative memory management it takes only one misbehaved program to crash the system.

Memory protection enables the kernel to limit a process' access to the computer's memory. Various methods of memory protection exist, including memory segmentation and paging. All methods require some level of hardware support (such as the 80286 MMU) which doesn't exist in all computers.

In both segmentation and paging, certain protected mode registers specify to the CPU what memory address it should allow a running program to access. Attempts to access other addresses will trigger an interrupt which will cause the CPU to re-enter supervisor mode, placing the kernel in charge. This is called a segmentation violation or Seg-V for short, and since it is both difficult to assign a meaningful result to such an operation, and because it is usually a sign of a misbehaving program, the kernel will generally resort to terminating the offending program, and will report the error.

Windows 3.1-Me had some level of memory protection, but programs could easily circumvent the need to use it. Under Windows 9x all MS-DOS applications ran in supervisor mode, giving them almost unlimited control over the computer. A general protection fault would be produced indicating a segmentation violation had occurred, however the system would often crash anyway.

In most Linux systems, part of the hard disk is reserved for virtual memory when the Operating system is being installed on the system. This part is known as swap space. Windows systems use a swap file instead of a partition.

Virtual memory

The use of virtual memory addressing (such as paging or segmentation) means that the kernel can choose what memory each program may use at any given time, allowing the operating system to use the same memory locations for multiple tasks.

If a program tries to access memory that isn't in its current range of accessible memory, but nonetheless has been allocated to it, the kernel will be interrupted in the same way as it would if the program were to exceed its allocated memory. (See section on memory management.) Under UNIX this kind of interrupt is referred to as a page fault.

When the kernel detects a page fault it will generally adjust the virtual memory range of the program

which triggered it, granting it access to the memory requested. This gives the kernel discretionary power over where a particular application's memory is stored, or even whether or not it has actually been allocated yet.

In modern operating systems, memory which is accessed less frequently can be temporarily stored on disk or other media to make that space available for use by other programs. This is called swapping, as an area of memory can be used by multiple programs, and what that memory area contains can be swapped or exchanged on demand.

Further information: Page fault

Multitasking

Main article: Computer multitasking

Main article: Process management (computing)

Multitasking refers to the running of multiple independent computer programs on the same computer, giving the appearance that it is performing the tasks at the same time. Since most computers can do at most one or two things at one time, this is generally done via time sharing, which means that each program uses a share of the computer's time to execute.

An operating system kernel contains a piece of software called a scheduler which determines how much time each program will spend executing, and in which order execution control should be passed to programs. Control is passed to a process by the kernel, which allows the program access to the CPU and memory. At a later time control is returned to the kernel through some mechanism, so that another program may be allowed to use the CPU. This so-called passing of control between the kernel and applications is called a context switch.

An early model which governed the allocation of time to programs was called cooperative multitasking. In this model, when control is passed to a program by the kernel, it may execute for as long as it wants before explicitly returning control to the kernel. This means that a malicious or malfunctioning program may not only prevent any other programs from using the CPU, but it can hang the entire system if it enters an infinite loop.

The philosophy governing preemptive multitasking is that of ensuring that all programs are given regular time on the CPU. This implies that all programs must be limited in how much time they are allowed to spend on the CPU without being interrupted. To accomplish this, modern operating system kernels make use of a timed interrupt. A protected mode timer is set by the kernel which triggers a return to supervisor mode after the specified time has elapsed. (See above sections on Interrupts and Dual Mode Operation.)

On many single user operating systems cooperative multitasking is perfectly adequate, as home computers generally run a small number of well tested programs. Windows NT was the first version of Microsoft Windows which enforced preemptive multitasking, but it didn't reach the home user market until Windows XP, (since Windows NT was targeted at professionals.)

Further information: Context switch

Further information: Preemptive multitasking

Further information: Cooperative multitasking

Kernel preemption

In recent years, concerns have arisen because of long latencies associated with some kernel run-times, sometimes on the order of 100ms or more in systems with monolithic kernels. These latencies often produce noticeable slowness in desktop systems, and can prevent operating systems from performing time-sensitive operations such as audio recording and some communications.[4]

Modern operating systems extend the concepts of application preemption to device drivers and kernel code, so that the operating system has preemptive control over internal run-times as well. Under Windows Vista, the introduction of the Windows Display Driver Model (WDDM) accomplishes this for display drivers, and in Linux, the preemptable kernel model introduced in version 2.6 allows all device drivers and some other parts of kernel code to take advantage of preemptive multi-tasking.

Under Windows prior to Windows Vista and Linux prior to version 2.6 all driver execution was co-operative, meaning that if a driver entered an infinite loop it would freeze the system.

Disk access and file systems

Main article: Virtual file system

Access to data stored on disks is a central feature of all operating systems. Computers store data on disks using files, which are structured in specific ways in order to allow for faster access, higher reliability, and to make better use out of the drive's available space. The specific way in which files are stored on a disk is called a file system, and enables files to have names and attributes. It also allows them to be stored in a hierarchy of directories or folders arranged in a directory tree.

Early operating systems generally supported a single type of disk drive and only one kind of file system. Early file systems were limited in their capacity, speed, and in the kinds of file names and directory structures they could use. These limitations often reflected limitations in the operating systems they were designed for, making it very difficult for an operating system to support more than one file system.

While many simpler operating systems support a limited range of options for accessing storage systems, operating systems like UNIX and Linux support a technology known as a virtual file system or VFS. An operating system like UNIX supports a wide array of storage devices, regardless of their design or file systems to be accessed through a common application programming interface (API). This makes it unnecessary for programs to have any knowledge about the device they are accessing. A VFS allows the operating system to provide programs with access to an unlimited number of devices with an infinite variety of file systems installed on them through the use of specific device drivers and file system drivers.

A connected storage device such as a hard drive is accessed through a device driver. The device driver understands the specific language of the drive and is able to translate that language into a standard language used by the operating system to access all disk drives. On UNIX this is the language of block devices.

When the kernel has an appropriate device driver in place, it can then access the contents of the disk drive in raw format, which may contain one or more file systems. A file system driver is used to translate the commands used to access each specific file system into a standard set of commands that the operating system can use to talk to all file systems. Programs can then deal with these file systems on the basis of filenames, and directories/folders, contained within a hierarchical structure. They can create, delete, open, and close files, as well as gather various information about them, including access

permissions, size, free space, and creation and modification dates.

Various differences between file systems make supporting all file systems difficult. Allowed characters in file names, case sensitivity, and the presence of various kinds of file attributes makes the implementation of a single interface for every file system a daunting task. Operating systems tend to recommend the use of (and so support natively) file systems specifically designed for them; for example, NTFS in Windows and ext3 and ReiserFS in Linux. However, in practice, third party drives are usually available to give support for the most widely used filesystems in most general-purpose operating systems (for example, NTFS is available in Linux through NTFS-3g, and ext2/3 and ReiserFS are available in Windows through FS-driver (http://www.fs-driver.org/) and rfstool (http://p-nand-q.com/download/rfstool.html)).

Device drivers

Main article: Device driver

A device driver is a specific type of computer software developed to allow interaction with hardware devices. Typically this constitutes an interface for communicating with the device, through the specific computer bus or communications subsystem that the hardware is connected to, providing commands to and/or receiving data from the device, and on the other end, the requisite interfaces to the operating system and software applications. It is a specialized hardware-dependent computer program which is also operating system specific that enables another program, typically an operating system or applications software package or computer program running under the operating system kernel, to interact transparently with a hardware device, and usually provides the requisite interrupt handling necessary for any necessary asynchronous time-dependent hardware interfacing needs.

The key design goal of device drivers is abstraction. Every model of hardware (even within the same class of device) is different. Newer models also are released by manufacturers that provide more reliable or better performance and these newer models are often controlled differently. Computers and their operating systems cannot be expected to know how to control every device, both now and in the future. To solve this problem, OSes essentially dictate how every type of device should be controlled. The function of the device driver is then to translate these OS mandated function calls into device specific calls. In theory a new device, which is controlled in a new manner, should function correctly if a suitable driver is available. This new driver will ensure that the device appears to operate as usual from the operating systems' point of view.

Networking

Main article: Computer network

Currently most operating systems support a variety of networking protocols, hardware, and applications for using them. This means that computers running dissimilar operating systems can participate in a common network for sharing resources such as computing, files, printers, and scanners using either wired or wireless connections. Networks can essentially allow a computer's operating system to access the resources of a remote computer to support the same functions as it could if those resources were connected directly to the local computer. This includes everything from simple communication, to using networked file systems or even sharing another computer's graphics or sound hardware. Some network services allow the resources of a computer to be accessed transparently, such as SSH which allows networked users direct access to a computer's command line interface.

Client/server networking involves a program on a computer somewhere which connects via a network to

another computer, called a server. Servers, usually running UNIX or Linux, offer (or host) various services to other network computers and users. These services are usually provided through ports or numbered access points beyond the server's network address. Each port number is usually associated with a maximum of one running program, which is responsible for handling requests to that port. A daemon, being a user program, can in turn access the local hardware resources of that computer by passing requests to the operating system kernel.

Many operating systems support one or more vendor-specific or open networking protocols as well, for example, SNA on IBM systems, DECnet on systems from Digital Equipment Corporation, and Microsoft-specific protocols (SMB) on Windows. Specific protocols for specific tasks may also be supported such as NFS for file access. Protocols like ESound, or esd can be easily extended over the network to provide sound from local applications, on a remote system's sound hardware.

Security

Main article: Computer security

A computer being secure depends on a number of technologies working properly. A modern operating system provides access to a number of resources, which are available to software running on the system, and to external devices like networks via the kernel.

The operating system must be capable of distinguishing between requests which should be allowed to be processed, and others which should not be processed. While some systems may simply distinguish between "privileged" and "non-privileged", systems commonly have a form of requester *identity*, such as a user name. To establish identity there may be a process of *authentication*. Often a username must be quoted, and each username may have a password. Other methods of authentication, such as magnetic cards or biometric data, might be used instead. In some cases, especially connections from the network, resources may be accessed with no authentication at all (such as reading files over a network share). Also covered by the concept of requester **identity** is *authorization*; the particular services and resources accessible by the requester once logged into a system and tied to either the requester's user account or to the variously configured groups of users to which the requester belongs.

In addition to the allow/disallow model of security, a system with a high level of security will also offer auditing options. These would allow tracking of requests for access to resources (such as, "who has been reading this file?"). Internal security, or security from an already running program is only possible if all possibly harmful requests must be carried out through interrupts to the operating system kernel. If programs can directly access hardware and resources, they cannot be secured.

External security involves a request from outside the computer, such as a login at a connected console or some kind of network connection. External requests are often passed through device drivers to the operating system's kernel, where they can be passed onto applications, or carried out directly. Security of operating systems has long been a concern because of highly sensitive data held on computers, both of a commercial and military nature. The United States Government Department of Defense (DoD) created the *Trusted Computer System Evaluation Criteria* (TCSEC) which is a standard that sets basic requirements for assessing the effectiveness of security. This became of vital importance to operating system makers, because the TCSEC was used to evaluate, classify and select computer systems being considered for the processing, storage and retrieval of sensitive or classified information.

Network services include offerings such as file sharing, print services, email, web sites, and file transfer protocols (FTP), most of which can have compromised security. At the front line of security are hardware devices known as firewalls or intrusion detection/prevention systems. At the operating system

level, there are a number of software firewalls available, as well as intrusion detection/prevention systems. Most modern operating systems include a software firewall, which is enabled by default. A software firewall can be configured to allow or deny network traffic to or from a service or application running on the operating system. Therefore, one can install and be running an insecure service, such as Telnet or FTP, and not have to be threatened by a security breach because the firewall would deny all traffic trying to connect to the service on that port.

An alternative strategy, and the only sandbox strategy available in systems that do not meet the Popek and Goldberg virtualization requirements, is the operating system not running user programs as native code, but instead either emulates a processor or provides a host for a p-code based system such as Java.

Internal security is especially relevant for multi-user systems; it allows each user of the system to have private files that the other users cannot tamper with or read. Internal security is also vital if auditing is to be of any use, since a program can potentially bypass the operating system, inclusive of bypassing auditing.

Example: Microsoft Windows

While the Windows 9x series offered the option of having profiles for multiple users, they had no concept of access privileges, and did not allow concurrent access; and so were not true multi-user operating systems. In addition, they implemented only partial memory protection. They were accordingly widely criticised for lack of security.

The Windows NT series of operating systems, by contrast, are true multi-user, and implement absolute memory protection. However, a lot of the advantages of being a true multi-user operating system were nullified by the fact that, prior to Windows Vista, the first user account created during the setup process was an administrator account, which was also the default for new accounts. Though Windows XP did have limited accounts, the majority of home users did not change to an account type with fewer rights – partially due to the number of programs which unnecessarily required administrator rights – and so most home users ran as administrator all the time.

Windows Vista changes this[5] by introducing a privilege elevation system called User Account Control. When logging in as a standard user, a logon session is created and a token containing only the most basic privileges is assigned. In this way, the new logon session is incapable of making changes that would affect the entire system. When logging in as a user in the Administrators group, two separate tokens are assigned. The first token contains all privileges typically awarded to an administrator, and the second is a restricted token similar to what a standard user would receive. User applications, including the Windows Shell, are then started with the restricted token, resulting in a reduced privilege environment even under an Administrator account. When an application requests higher privileges or "Run as administrator" is clicked, UAC will prompt for confirmation and, if consent is given (including administrator credentials if the account requesting the elevation is not a member of the administrators group), start the process using the unrestricted token.[6]

Example: Linux/Unix

Linux and UNIX both have two tier security, which limits any system-wide changes to the root user, a special user account on all UNIX-like systems. While the root user has virtually unlimited permission to effect system changes, programs running as a regular user are limited in where they can save files, what hardware they can access, etc. In many systems, a user's memory usage, their selection of available programs, their total disk usage or quota, available range of programs' priority settings, and other functions can also be locked down. This provides the user with plenty of freedom to do what needs to be

done, without being able to put any part of the system in jeopardy (barring accidental triggering of system-level bugs) or make sweeping, system-wide changes. The user's settings are stored in an area of the computer's file system called the user's home directory, which is also provided as a location where the user may store their work, a concept later adopted by Windows as the 'My Documents' folder. Should a user have to install software outside of his home directory or make system-wide changes, they must become the root user temporarily, usually with the `su` or sudo command, which is answered with the computer's root password when prompted. Some systems (such as Ubuntu and its derivatives) are configured by default to allow select users to run programs as the root user via the `sudo` command, using the user's own password for authentication instead of the system's root password. One is sometimes said to "go root" or "drop to root" when elevating oneself to root access.

For more information on the differences between the Linux `su`/`sudo` approach and Vista's User Account Control, see Comparison of privilege authorization features.

File system support in modern operating systems

Support for file systems is highly varied among modern operating systems although there are several common file systems which almost all operating systems include support and drivers for.

Solaris

The Solaris Operating System (as with most operating systems based upon open standards and/or open source) uses UFS as its primary file system. Prior to 1998, Solaris UFS did not have logging/journaling capabilities, but over time the OS has gained this and other new data management capabilities.

Additional features include Veritas (Journaling) VxFS, QFS from Sun Microsystems, enhancements to UFS including multiterabyte support and UFS volume management included as part of the OS, and ZFS (open source, poolable, 128-bit, compressible, and error-correcting).

Kernel extensions were added to Solaris to allow for bootable Veritas VxFS operation. Logging or journaling was added to UFS in Solaris 7. Releases of Solaris 10, Solaris Express, OpenSolaris, and other open source variants of Solaris later supported bootable ZFS.

Logical Volume Management allows for spanning a file system across multiple devices for the purpose of adding redundancy, capacity, and/or throughput. Solaris includes Solaris Volume Manager (formerly known as Solstice DiskSuite.) Solaris is one of many operating systems supported by Veritas Volume Manager. Modern Solaris based operating systems eclipse the need for volume management through leveraging virtual storage pools in ZFS.

Linux

Many Linux distributions support some or all of ext2, ext3, ext4, ReiserFS, Reiser4, JFS , XFS , GFS, GFS2, OCFS, OCFS2, and NILFS. The ext file systems, namely ext2, ext3 and ext4 are based on the original Linux file system. Others have been developed by companies to meet their specific needs, hobbyists, or adapted from UNIX, Microsoft Windows, and other operating systems. Linux has full support for XFS and JFS, along with FAT (the MS-DOS file system), and HFS which is the primary file system for the Macintosh.

In recent years support for Microsoft Windows NT's NTFS file system has appeared in Linux, and is now comparable to the support available for other native UNIX file systems. ISO 9660 and Universal Disk Format (UDF) are supported which are standard file systems used on CDs, DVDs, and BluRay

discs. It is possible to install Linux on the majority of these file systems. Unlike other operating systems, Linux and UNIX allow any file system to be used regardless of the media it is stored in, whether it is a hard drive, a disc (CD,DVD...), an USB key, or even contained within a file located on another file system.

Microsoft Windows

Microsoft Windows currently supports NTFS and FAT file systems, along with network file systems shared from other computers, and the ISO 9660 and UDF filesystems used for CDs, DVDs, and other optical discs such as Blu-ray. Under Windows each file system is usually limited in application to certain media, for example CDs must use ISO 9660 or UDF, and as of Windows Vista, NTFS is the only file system which the operating system can be installed on. Windows Embedded CE 6.0, Windows Vista Service Pack 1, and Windows Server 2008 support ExFAT, a file system more suitable for flash drives.

Mac OS X

Mac OS X supports HFS+ with journaling as its primary file system. It is derived from the Hierarchical File System of the earlier Mac OS. Mac OS X has facilities to read and write FAT, NTFS (read-only, although an open-source cross platform implementation known as NTFS 3G provides read-write support to Microsoft Windows NTFS file system for Mac OS X users), UDF, and other file systems, but cannot be installed to them. Due to its UNIX heritage Mac OS X now supports virtually all the file systems supported by the UNIX VFS. Recently Apple Inc. started work on porting Sun Microsystems' ZFS filesystem to Mac OS X and preliminary support is already available in Mac OS X 10.5 as well as support for Blu-ray discs.

Special-purpose file systems

FAT file systems are commonly found on floppy disks, flash memory cards, digital cameras, and many other portable devices because of their relative simplicity. Performance of FAT compares poorly to most other file systems as it uses overly simplistic data structures, making file operations time-consuming, and makes poor use of disk space in situations where many small files are present. ISO 9660 and Universal Disk Format are two common formats that target Compact Discs and DVDs. Mount Rainier is a newer extension to UDF supported by Linux 2.6 series and Windows Vista that facilitates rewriting to DVDs in the same fashion as has been possible with floppy disks.

Journalized file systems

File systems may provide journaling, which provides safe recovery in the event of a system crash. A journaled file system writes some information twice: first to the journal, which is a log of file system operations, then to its proper place in the ordinary file system. Journaling is handled by the file system driver, and keeps track of each operation taking place that changes the contents of the disk. In the event of a crash, the system can recover to a consistent state by replaying a portion of the journal. Many UNIX file systems provide journaling including ReiserFS, JFS, and Ext3.

In contrast, non-journaled file systems typically need to be examined in their entirety by a utility such as fsck or chkdsk for any inconsistencies after an unclean shutdown. Soft updates is an alternative to journaling that avoids the redundant writes by carefully ordering the update operations. Log-structured file systems and ZFS also differ from traditional journaled file systems in that they avoid inconsistencies by always writing new copies of the data, eschewing in-place updates.

Graphical user interfaces

Most of the modern computer systems support graphical user interfaces (GUI), and often include them. In some computer systems, such as the original implementations of Microsoft Windows and the Mac OS, the GUI is integrated into the kernel.

While technically a graphical user interface is not an operating system service, incorporating support for one into the operating system kernel can allow the GUI to be more responsive by reducing the number of context switches required for the GUI to perform its output functions. Other operating systems are modular, separating the graphics subsystem from the kernel and the Operating System. In the 1980s UNIX, VMS and many others had operating systems that were built this way. Linux and Mac OS X are also built this way. Modern releases of Microsoft Windows such as Windows Vista implement a graphics subsystem that is mostly in user-space, however versions between Windows NT 4.0 and Windows Server 2003's graphics drawing routines exist mostly in kernel space. Windows 9x had very little distinction between the interface and the kernel.

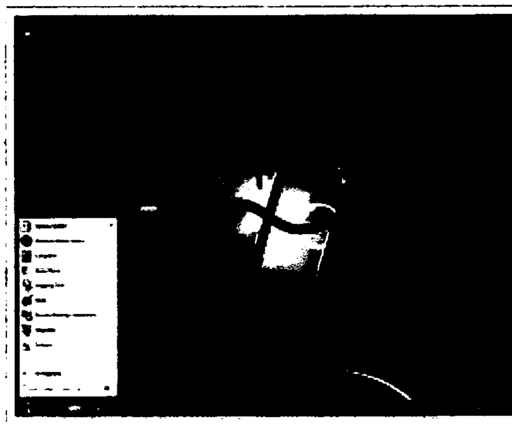
Many computer operating systems allow the user to install or create any user interface they desire. The X Window System in conjunction with GNOME or KDE is a commonly-found setup on most Unix and Unix-like (BSD, Linux, Minix) systems. A number of Windows shell replacements have been released for Microsoft Windows, which offer alternatives to the included Windows shell, but the shell itself cannot be separated from Windows.

Numerous Unix-based GUIs have existed over time, most derived from X11. Competition among the various vendors of Unix (HP, IBM, Sun) led to much fragmentation, though an effort to standardize in the 1990s to COSE and CDE failed for the most part due to various reasons, eventually eclipsed by the widespread adoption of GNOME and KDE. Prior to open source-based toolkits and desktop environments, Motif was the prevalent toolkit/desktop combination (and was the basis upon which CDE was developed).

Graphical user interfaces evolve over time. For example, Windows has modified its user interface almost every time a new major version of Windows is released, and the Mac OS GUI changed dramatically with the introduction of Mac OS X in 1999.[7]

Examples of operating systems

Microsoft Windows



Windows 7 is the latest stable Windows

Microsoft Windows is a family of proprietary operating systems that originated as an add-on to the older MS-DOS operating system for the IBM PC. Modern versions are based on the newer Windows NT kernel that was originally intended for OS/2. Windows runs on x86, x86-64 and Itanium processors. Earlier versions also ran on the Alpha, MIPS, Fairchild (later Intergraph) Clipper and PowerPC architectures (some work was done to port it to the SPARC architecture).

As of June 2008, Microsoft Windows holds a large amount of the worldwide desktop market share. Windows is also used on servers, supporting applications such as web servers

operating system.

and database servers. In recent years, Microsoft has spent significant marketing and research & development money to demonstrate that Windows is capable of running any enterprise application, which has resulted in consistent price/performance records (see the TPC) and significant acceptance in the enterprise market.

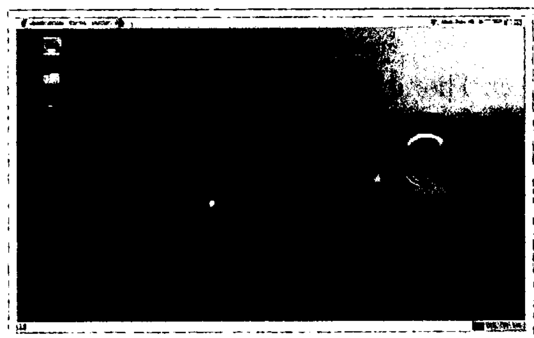
The most widely used version of the Microsoft Windows family is Windows XP, released on October 25, 2001.

In November 2006, after more than five years of development work, Microsoft released Windows Vista, a major new operating system version of Microsoft Windows family which contains a large number of new features and architectural changes. Chief amongst these are a new user interface and visual style called Windows Aero, a number of new security features such as User Account Control, and a few new multimedia applications such as Windows DVD Maker. A server variant based on the same kernel, Windows Server 2008, was released in early 2008.

On October 22, 2009, Microsoft released Windows 7, the successor to Windows Vista, coming three years after its release. While Vista was about introducting new features, Windows 7 aims to streamline these and provide for a faster overall working environment. Windows Server 2008 R2, the server variant, was released at the same time.

Unix and Unix-like operating systems

Ken Thompson wrote B, mainly based on BCPL, which he used to write Unix, based on his experience in the MULTICS project. B was replaced by C, and Unix developed into a large, complex family of inter-related operating systems which have been influential in every modern operating system (see History). The *Unix-like* family is a diverse group of operating systems, with several major sub-categories including System V, BSD, and Linux. The name "UNIX" is a trademark of The Open Group which licenses it for use with any operating system that has been shown to conform to their definitions. "Unix-like" is commonly used to refer to the large set of operating systems which resemble the original Unix.


Debian is a (linux-based) unix-like system

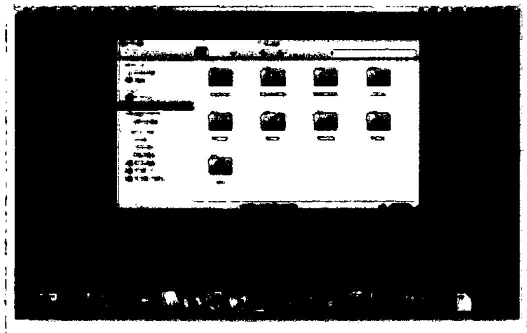
Unix-like systems run on a wide variety of machine architectures. They are used heavily for servers in business, as well as workstations in academic and engineering environments. Free Unix variants, such as GNU, Linux and BSD, are popular in these areas.

Some Unix variants like HP's HP-UX and IBM's AIX are designed to run only on that vendor's hardware. Others, such as Solaris, can run on multiple types of hardware, including x86 servers and PCs. Apple's Mac OS X, a hybrid kernel-based BSD variant derived from NeXTSTEP, Mach, and FreeBSD, has replaced Apple's earlier (non-Unix) Mac OS.

Unix interoperability was sought by establishing the POSIX standard. The POSIX standard can be applied to any operating system, although it was originally created for various Unix variants.

Mac OS X

Mac OS X is a line of partially proprietary, graphical operating systems developed, marketed, and sold by Apple Inc., the latest of which is pre-loaded on all currently shipping Macintosh computers. Mac OS X is the successor to the original Mac OS, which had been Apple's primary operating system since 1984. Unlike its predecessor, Mac OS X is a UNIX operating system built on technology that had been developed at NeXT through the second half of the 1980s and up until Apple purchased the company in early 1997.


Mac OS X Snow Leopard

The operating system was first released in 1999 as Mac OS X Server 1.0, with a desktop-oriented version (Mac OS X v10.0) following in March 2001. Since then, six more distinct "client" and "server" editions of Mac OS X have been released, the most recent being Mac OS X v10.6, which was first made available on August 28, 2009. Releases of Mac OS X are named after big cats; the current version of Mac OS X is nicknamed "Snow Leopard".

The server edition, Mac OS X Server, is architecturally identical to its desktop counterpart but usually runs on Apple's line of Macintosh server hardware. Mac OS X Server includes work group management and administration software tools that provide simplified access to key network services, including a mail transfer agent, a Samba server, an LDAP server, a domain name server, and others.

Plan 9

Ken Thompson, Dennis Ritchie and Douglas McIlroy at Bell Labs designed and developed the C programming language to build the operating system Unix. Programmers at Bell Labs went on to develop Plan 9 and Inferno, which were engineered for modern distributed environments. Plan 9 was designed from the start to be a networked operating system, and had graphics built-in, unlike Unix, which added these features to the design later. Plan 9 has yet to become as popular as Unix derivatives, but it has an expanding community of developers. It is currently released under the Lucent Public License. Inferno was sold to Vita Nuova Holdings and has been released under a GPL/MIT license.

Real-time operating systems

Main article: real-time operating system

A real-time operating system (RTOS1) is a multitasking operating system intended for applications with fixed deadlines (real-time computing). Such applications include some small embedded systems, automobile engine controllers, industrial robots, spacecraft, industrial control, and some large-scale computing systems.

An early example of a large-scale real-time operating system was Transaction Processing Facility developed by American Airlines and IBM for the Sabre Airline Reservations System.

Embedded systems that have fixed deadlines use a real-time operating system such as VxWorks, eCos, QNX, MontaVista Linux and RTLinux. Windows CE is a real-time operating system that shares similar APIs to desktop Windows but shares none of desktop Windows' codebase.

Some embedded systems use operating systems such as Symbian OS, Palm OS, BSD, and Linux, although such operating systems do not support real-time computing.

Hobby development

Operating system development, or OSDev for short, as a hobby has a large cult-like following. As such, operating systems, such as Linux, have derived from hobby operating system projects. The design and implementation of an operating system requires skill and determination, and the term can cover anything from a basic "Hello World" boot loader to a fully featured kernel. One classical example of this is the Minix Operating System—an OS that was designed by A.S. Tanenbaum as a teaching tool but was heavily used by hobbyists before Linux eclipsed it in popularity.

Other

Older operating systems which are still used in niche markets include OS/2 from IBM and Microsoft; Mac OS, the non-Unix precursor to Apple's Mac OS X; BeOS; XTS-300. Some, most notably AmigaOS 4 and RISC OS, continue to be developed as minority platforms for enthusiast communities and specialist applications. OpenVMS formerly from DEC, is still under active development by Hewlett-Packard.

There were a number of operating systems for 8 bit computers - Apple's DOS (Disk Operating System) 3.2 & 3.3 for Apple II, ProDOS, UCSD, CP/M - available for various 8 and 16 bit environments, FutureOS for the Amstrad CPC6128 and 6128Plus.

Research and development of new operating systems continues. GNU Hurd is designed to be backwards compatible with Unix, but with enhanced functionality and a microkernel architecture. Singularity is a project at Microsoft Research to develop an operating system with better memory protection based on the .Net managed code model. Systems development follows the same model used by other Software development, which involves maintainers, version control "trees", forks, "patches", and specifications. From the AT&T-Berkeley lawsuit the new unencumbered systems were based on 4.4BSD which forked as FreeBSD and NetBSD efforts to replace missing code after the Unix wars. Recent forks include DragonFly BSD and Darwin from BSD Unix.

Diversity of operating systems and portability

Application software is generally written for use on a specific operating system, and sometimes even for specific hardware. When porting the application to run on another OS, the functionality required by that application may be implemented differently by that OS (the names of functions, meaning of arguments, etc.) requiring the application to be adapted.

This cost in supporting operating systems diversity can be avoided by instead writing applications against software platforms like Java, Qt or for web browsers. These abstractions have already borne the cost of adaptation to specific operating systems and their system libraries.

Another approach is for operating system vendors to adopt standards. For example, POSIX and OS abstraction layers provide commonalities that reduce porting costs.

See also

- List of operating systems
- Comparison of operating systems
- Computer systems architecture
- Disk operating system

- Kernel (computer science)
- List of important publications in computer science#Operating systems
- Object-oriented operating system
- Orthogonal persistence
- Process management (computing)
- System call
- System image
- Trusted operating system

References

1. ^ "Dishwasher operating system (http://www.eurotechappliances.com/products/specs-dw.htm)". Eurotechappliances.com. http://www.eurotechappliances.com/products/specs-dw.htm. Retrieved 2009-07-15.
2. ^ Paul, Ryan (2007-09-03). "Linux market share set to surpass Win 98, OS X still ahead of Vista (http://arstechnica.com/news.ars/post/20070903-linux-marketshare-set-to-surpass-windows-98.html)". Arstechnica.com. http://arstechnica.com/news.ars/post/20070903-linux-marketshare-set-to-surpass-windows-98.html. Retrieved 2009-10-23.
3. ^ "Linux server market share keeps growing (http://www.linux-watch.com/news/NS5369154346.html)". Linux-watch.com. 2007-05-29. http://www.linux-watch.com/news/NS5369154346.html. Retrieved 2009-07-15.
4. ^ http://inf3-www.informatik.unibw-muenchen.de/research/linux/milan/paper_milan.pdf
5. ^ Microsoft describes in detail the steps taken to combat this in a TechNet bulletin. [1] (http://technet.microsoft.com/en-us/windowsvista/aa905073.aspx)
6. ^ Kenny Kerr (2006-09-29). "Windows Vista for Developers – Part 4 – User Account Control (http://weblogs.asp.net/kennykerr/archive/2006/09/29/Windows-Vista-for-Developers-_1320_-Part-4-_1320_-User-Account-Control.aspx)". http://weblogs.asp.net/kennykerr/archive/2006/09/29/Windows-Vista-for-Developers-_1320_-Part-4-_1320_-User-Account-Control.aspx. Retrieved 2007-03-15.
7. ^ Poisson, Ken. "Chronology of Personal Computer Software" (http://www.islandnet.com/~kpolsson/compsoft/soft1998.htm). Retrieved on 2008-05-07. Last checked on 2009-03-30.

- Auslander, Marc A.; Larkin, David C.; Scherr, Allan L. (1981). *The evolution of the MVS Operating System*. IBM J. Research & Development. http://www.research.ibm.com/journal/rd/255/auslander.pdf
- Deitel, Harvey M.; Deitel, Paul; Choffnes, David. *Operating Systems*. Pearson/Prentice Hall. ISBN 978-0-13-092641-8.
- Bic, Lubomur F.; Shaw, Alan C. (2003). *Operating Systems.*. Pearson: Prentice Hall.
- Stallings (2005). *Operating Systems, Internals and Design Principles*. Pearson: Prentice Hall.
- Silberschatz, Avi; Galvin, Peter; Gagne, Greg (2008). *Operating Systems Concepts*. John Wiley & Sons. ISBN 0-470-12872-0.

External links

- Operating Systems (http://www.dmoz.org/Computers/Software/Operating_Systems/) at the Open Directory Project
- UC Berkeley Video Course (http://academicearth.org/courses/operating-systems-and-system-programming) on Operating Systems
- Multics History (http://www.cbi.umn.edu/iterations/haigh.html) and the history of operating systems
- How Stuff Works - Operating Systems (http://computer.howstuffworks.com/operating-system.htm)

Categories: Operating systems

Chapter 16

How Bastion Hosts Work

CONTENTS

One of the best ways to protect an intranet from attack is to put a heavily fortified *bastion host* or *bastion server* in a firewall. Having a bastion host means that all access to an intranet from the Internet will be required to come through the bastion host. By concentrating all access in a single server, or a small group of servers, it's much easier to protect the entire intranet.

The bastion host does not provide intranet services itself. When it receives a request from the Internet for an intranet service, the host passes the request to the appropriate server. Subsequently, it takes the response and passes it back to the Internet.

Proxy server programs can also run on bastion hosts. That is, when someone on the intranet wants to get at an Internet resource, they first contact the proxy server on the bastion host, and the bastion host then relays the request to the Internet server. The Internet server sends the information to the proxy server on the bastion host, which in turn passes the information back to the user on the intranet.

Several means are taken to ensure that the bastion host is as secure as possible-and also to make sure that if the host is hacked into, intranet security won't be compromised.

To make the bastion host secure, it is stripped of all but the most basic services. A typical network server provides login, file, print, and other services, including access to additional servers. On a bastion host, those services have been prohibited. Since there are no user accounts, it's difficult for someone to break in using passwords. Since it has few services available, even if someone did break in, there wouldn't be much they could do with it.

For even more security, bastion hosts can be put on a private subnet (often referred to as a *perimeter network*), further isolating the host so that if someone breaks into it, they can only get access to that subnet, not to the rest of the intranet. A filtering router reviews packets coming from the private subnet, making sure that only authorized incoming requests pass through to the intranet.

Even more security measures can protect the server and intranet, sending alerts to intranet administrators if someone is trying to break in. The bastion host can log all access to it, and keep a secure backup of that log on a physically separate machine connected by the serial port so no one can gain access to the log remotely. System administrators can examine the log for signs of break-ins. Even more powerful are monitoring programs that watch the log and sound an alarm if it detects someone has been trying to break into the server. Auditing software can also constantly check the server software to see if it has been altered in any way-a possible sign that an intruder has successfully attacked it and taken control of its resources.

How Bastion Hosts Work

A bastion host (also called a bastion server) is one of the main defenses in an intranet firewall. It's a heavily fortified server that sits inside the firewall, and it is the main point of contact between the intranet and the Internet. By having an isolated, heavily defended server as the main point of contact, the rest of the intranet resources can be shielded from attacks starting on the Internet.

- Bastion hosts are built so that every network service possible is disabled on them-the only thing the server does is allow for specified Internet access. So, for example, there should be no user accounts on a bastion server, so that no one can log into it and take control of it and then gain access to the intranet. Even the Network File System (NFS), which allows a system to access files across a network on a remote system, should be disabled, so that intruders can't gain access to the bastion server and then get at files on the intranet. The safest way to use bastion hosts is to put them on their own subnet as part of an intranet firewall. By putting them on their own network, if they are broken into, no other intranet resources are compromised.
- Bastion servers log all activity so that intranet administrators can tell if the intranet has been attacked. They often keep two copies of system logs for security reasons: In case one log is destroyed or tampered with, the other log is always available as a backup. One way to keep a secure copy of the log is to connect the bastion server via a serial port to a dedicated computer, whose only purpose is to keep track of the secure backup log.
- Automated monitors are even more sophisticated programs than auditing software. Automated monitors regularly check the bastion server's system logs, and send an alarm if it finds a suspicious pattern. For example, an alarm might be sent if someone attempted more than three unsuccessful logins.
- There can be more than one bastion host in a firewall. Each bastion host can handle one or more Internet services for the intranet. Sometimes, a bastion host can be used as a victim machine. This is a server that is stripped bare of almost all services except one specific Internet service. Victim machines can be used to provide Internet services that are hard to handle using proxying or a filtering router, or whose security concerns are not yet known. The services are put on the victim machine instead of a bastion host with other services. That way, if the server is broken into, other bastion hosts won't be affected.
- Placing a filtering router between the bastion host and the intranet provides additional security. The filtering router checks all packets between the Internet and the intranet, dropping unauthorized traffic.
- When a bastion server receives a request for a service, such as sending a Web page or delivering e-mail, the server doesn't handle the request itself. Instead, it sends the request along to the appropriate intranet server. The intranet server handles the request, and then sends the information back to the bastion server. The bastion server now sends the requested information to the requester on the Internet.
- Some bastion servers include auditing programs, which actively check to see whether an attack has been launched against them. There are a variety of ways to do auditing. One way to audit is to use a checksum program, which checks to see whether any software on the bastion server has been changed by an unauthorized person. A checksum program calculates a number based on the size of an executable program on the server. It then regularly calculates the checksum to see if it has changed. If it has changed, someone has altered the software, which could signal an attack.

Distributed computing

From Wikipedia, the free encyclopedia

Distributed computing is a field of computer science that studies distributed systems. A **distributed system** consists of multiple autonomous computers that communicate through a computer network. The computers interact with each other in order to achieve a common goal. A computer program that runs in a distributed system is called a **distributed program**, and **distributed programming** is the process of writing such programs.[1]

Distributed computing also refers to the use of distributed systems to solve computational problems. In distributed computing, a problem is divided into many tasks, each of which is solved by one computer.[2]

Contents

Introduction

The word *distributed* in terms such as "distributed computing", "distributed system", "distributed programming", and "distributed algorithm" originally referred to computer networks where individual computers were physically distributed within some geographical area.[3] The terms are nowadays used in a much wider sense, even when referring to autonomous processes that run on the same physical computer and interact with each other by message passing.[4]

While there is no single definition of a distributed system,[5] the following defining properties are commonly used:

- There are several autonomous computational entities, each of which has its own local memory.[6]
- The entities communicate with each other by message passing.[7]

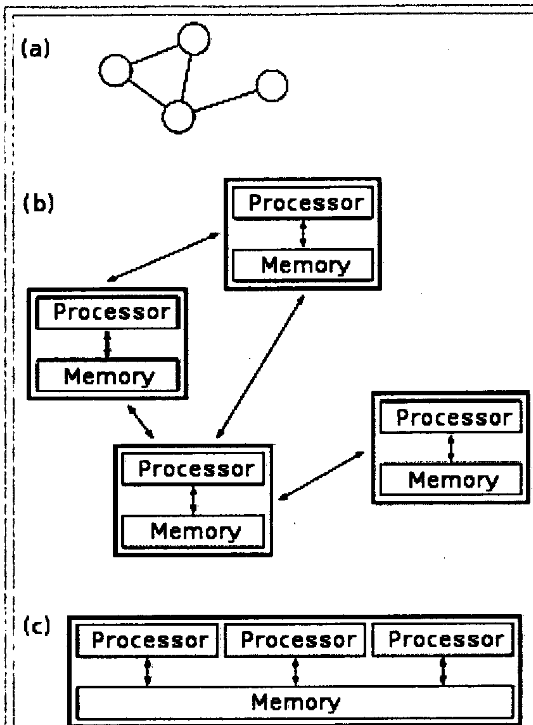
In this article, the computational entities are called *computers* or *nodes*.

A distributed system may have a common goal, such as solving a large computational problem.[8] Alternatively, each computer may have its own user with individual needs, and the purpose of the distributed system is to coordinate the use of shared resources or provide communication services to the users.[9]

Other typical properties of distributed systems include the following:

- The system has to tolerate failures in individual computers.[10]
- The structure of the system (network topology, network latency, number of computers) is not known in advance, the system may consist of different kinds of computers and network links, and the system may change during the execution of a distributed program.[11]
- Each computer has only a limited, incomplete view of the system. Each computer may know only one part of the input.[12]

Parallel or distributed computing?



(a)–(b) A distributed system.
(c) A parallel system.

The terms "concurrent computing", "parallel computing", and "distributed computing" have a lot of overlap, and no clear distinction exists between them.[13] The same system may be characterised both as "parallel" and "distributed"; the processors in a typical distributed system run concurrently in parallel.[14] Parallel computing may be seen as a particular tightly-coupled form of distributed computing,[15] and distributed computing may be seen as a loosely-coupled form of parallel computing.[16] Nevertheless, it is possible to roughly classify concurrent systems as "parallel" or "distributed" using the following criteria:

- In parallel computing, all processors have access to a shared memory. Shared memory can be used to exchange information between processors.[17]
- In distributed computing, each processor has its own private memory (distributed memory). Information is exchanged by passing messages between the processors.[18]

The figure on the right illustrates the difference between distributed and parallel systems. Figure (a) is a schematic view of a typical distributed system; as usual, the system is represented as a graph in which each node (vertex) is a computer and each edge (line between two nodes) is a communication link. Figure (b) shows the same distributed system in more detail: each computer has its own local memory, and information can be exchanged only by passing messages from one node to another by using the available communication links. Figure (c) shows a parallel system in which each processor has a direct access to a shared memory.

The situation is further complicated by the traditional uses of the terms parallel and distributed *algorithm* that do not quite match the above definitions of parallel and distributed *systems*; see the section Theoretical foundations below for more detailed discussion. Nevertheless, as a rule of thumb, high-performance parallel computation in a shared-memory multiprocessor uses parallel algorithms while the coordination of a large-scale distributed system uses distributed algorithms.

History

The use of concurrent processes that communicate by message-passing has its roots in operating system architectures studied in 1960s.[19] The first wide-spread distributed systems were local-area networks such as Ethernet that was invented in 1970s.[20]

ARPANET, the predecessor of the Internet, was introduced in the late 1960s, and ARPANET e-mail was invented in the early 1970s. E-mail became the most successful application of ARPANET,[21] and it is probably the earliest example of a large-scale distributed application. In addition to ARPANET and its successor Internet, other early worldwide computer networks included Usenet and FidoNet from 1980s, both of which were used to support distributed discussion systems.

The study of distributed computing became its own branch of computer science in the late 1970s and early 1980s. The first conference in the field, Symposium on Principles of Distributed Computing (PODC), dates back to 1982, and its European counterpart International Symposium on Distributed Computing (DISC) was first held in 1985.

Applications

There are two main reasons for using distributed systems and distributed computing. First, the very nature of the application may *require* the use of a communication network that connects several computers. For example, data is produced in one physical location and it is needed in another location.

Second, there are many cases in which the use of a single computer would be possible in principle, but the use of a distributed system is *beneficial* for practical reasons. For example, it may be more cost-efficient to obtain the desired level of performance by using a cluster of several low-end computers, in comparison with a single high-end computer. A distributed system can be more reliable than a non-distributed system, as there is no single point of failure. Moreover, a distributed system may be easier to expand and manage than a monolithic uniprocessor system.[22]

Examples of distributed systems and applications of distributed computing include the following:[23]

- Telecommunication networks:
  - Telephone networks and cellular networks.
  - Computer networks such as the Internet.
  - Wireless sensor networks.
  - Routing algorithms.
- Network applications:
  - World wide web and peer-to-peer networks.
  - Massively multiplayer online games and virtual reality communities.
  - Distributed databases and distributed database management systems.
  - Network file systems.
  - Distributed information processing systems such as banking systems and airline reservation systems.
- Real-time process control:
  - Aircraft control systems.
  - Industrial control systems.
- Parallel computation:
  - Scientific computing, including cluster computing and grid computing and various volunteer computing projects; see the list of distributed computing projects.

- Distributed rendering in computer graphics.

Theoretical foundations

Models

Many tasks that we would like to automate by using a computer are of question–answer type: we would like to ask a question and the computer should produce an answer. In theoretical computer science, such tasks are called computational problems. Formally, a computational problem consists of *instances* together with a *solution* for each instance. Instances are questions that we can ask, and solutions are desired answers to these questions.

Theoretical computer science seeks to understand which computational problems can be solved by using a computer (computability theory) and how efficiently (computational complexity theory). Traditionally, it is said that a problem can be solved by using a computer if we can design an algorithm that produces a correct solution for any given instance. Such an algorithm can be implemented as a computer program that runs on a general-purpose computer: the program reads a problem instance from input, performs some computation, and produces the solution as output. Formalisms such as random access machines or universal Turing machines can be used as abstract models of a sequential general-purpose computer executing such an algorithm.

The field of concurrent and distributed computing studies similar questions in the case of either multiple computers, or a computer that executes a network of interacting processes: which computational problems can be solved in such a network and how efficiently? However, it is not at all obvious what is meant by "solving a problem" in the case of a concurrent or distributed system: for example, what is the task of the algorithm designer, and what is the concurrent and/or distributed equivalent of a sequential general-purpose computer?

The discussion below focusses on the case of multiple computers, although many of the issues are the same for concurrent processes running on a single computer.

Three viewpoints are commonly used:

Parallel algorithms in shared-memory model

- All computers have access to a shared memory. The algorithm designer chooses the program executed by each computer.
- Models such as parallel random access machines (PRAM) are used.[24]

Parallel algorithms in message-passing model

- The algorithm designer chooses the structure of the network, as well as the program executed by each computer.
- Models such as Boolean circuits and sorting networks are used.[25] A Boolean circuit can be seen as a computer network: each gate is a computer that runs an extremely simple computer program. Similarly, a sorting network can be seen as a computer network: each comparator is a computer.

Distributed algorithms in message-passing model

- The algorithm designer only chooses the computer program. All computers run the same program. The system must work correctly regardless of the structure of the network.
- A commonly used model is a graph with one finite-state machine per node.

In the case of distributed algorithms, computational problems are typically related to graphs. Often the graph that describes the structure of the computer network *is* the problem instance. This is illustrated in the following example.

An example

Consider the computational problem of finding a coloring of a given graph G. Different fields might take the following approaches:

Centralized algorithms

- The graph G is encoded as a string, and the string is given as input to a computer. The computer program finds a coloring of the graph, encodes the coloring as a string, and outputs the result.

Parallel algorithms

- Again, the graph G is encoded as a string. However, multiple computers can access the same string in parallel. Each computer might focus on one part of the graph and produce a colouring for that part.
- The main focus is on high-performance computation that exploits the processing power of multiple computers in parallel.

Distributed algorithms

- The graph G is the structure of the computer network. There is one computer for each node of G and one communication link for each edge of G. Initially, each computer only knows about its immediate neighbours in the graph G; the computers must exchange messages with each other to discover more about the structure of G. Each computer must produce its own colour as output.
- The main focus is on coordinating the operation of an arbitrary distributed system.

While the field of parallel algorithms has a different focus than the field of distributed algorithms, there is a lot of interaction between the two fields. For example, the Cole–Vishkin algorithm for graph colouring[26] was originally presented as a parallel algorithm, but the same technique can also be used directly as a distributed algorithm.

Moreover, a parallel algorithm can be implemented either in a parallel system (using shared memory) or in a distributed system (using message passing).[27] The traditional boundary between parallel and distributed algorithms (choose a suitable network vs. run in any given network) does not lie in the same place as the boundary between parallel and distributed systems (shared memory vs. message passing).

Complexity measures

A centralised algorithm is efficient if it does not require much time (number of computational steps) or space (amount of memory). These complexity measures give rise to complexity classes such as P (decision problems solvable in polynomial time) and PSPACE (decision problems solvable in polynomial space).

In parallel algorithms, yet another resource in addition to time and space is the number of computers. Indeed, often there is a trade-off between the running time and the number of computers: the problem can be solved faster if there are more computers running in parallel (see speedup). If a decision problem can be solved in polylogarithmic time by using a polynomial number of processors, then the problem is said to be in the class

NC.[28] The class NC can be defined equally well by using the PRAM formalism or Boolean circuits – PRAM machines can simulate Boolean circuits efficiently and vice versa.[29]

In the analysis of distributed algorithms, more attention is usually paid on communication operations than computational steps. Perhaps the simplest model of distributed computing is a synchronous system where all nodes operate in a lockstep fashion. During each *communication round*, all nodes in parallel (1) receive the latest messages from their neighbours, (2) perform arbitrary local computation, and (3) send new messages to their neighbours. In such systems, a central complexity measure is the number of synchronous communication rounds required to complete the task.[30]

This complexity measure is closely related to the diameter of the network. Let D be the diameter of the network. On the one hand, any computable problem can be solved trivially in a synchronous distributed system in approximately $2D$ communication rounds: simply gather all information in one location (D rounds), solve the problem, and inform each node about the solution (D rounds).

On the other hand, if the running time of the algorithm is much smaller than D communication rounds, then the nodes in the network must produces their output without having the possibility to obtain information about distant parts of the network. In other words, the nodes must make globally consistent decisions based on information that is available in their *local neighbourhood*. Many distributed algorithms are known with the running time much smaller than D rounds, and understanding which problems can be solved by such algorithms is one of the central research questions of the field.[31]

Other commonly used measures are the total number of bits transmitted in the network (cf. communication complexity).

Other problems

Traditional computational problems take the perspective that we ask a question, a computer (or a distributed system) processes the question for a while, and then produces an answer and stops. However, there are also problems where we do not want the system to ever stop. Examples of such problems include the dining philosophers problem and other similar mutual exclusion problems. In these problems, the distributed system is supposed to continuously coordinate the use of shared resources so that no conflicts or deadlocks occur.

There are also fundamental challenges that are unique to distributed computing. The first example is challenges that are related to *fault-tolerance*. Examples of related problems include consensus problems,[32] Byzantine fault tolerance,[33] and self-stabilisation.[34]

A lot of research is also focused on understanding the *asynchronous* nature of distributed systems:

- Synchronizers can be used to run synchronous algorithms in asynchronous systems.[35]
- Logical clocks provide a causal happened-before ordering of events.[36]
- Clock synchronization algorithms provide globally consistent physical time stamps.[37]

Properties of distributed systems

So far the focus has been on *designing* a distributed system that solves a given problem. A complementary research problem is *studying* the properties of a given distributed system.

The halting problem is an analogous example from the field of centralised computation: we are given a computer program and the task is to decide whether it halts or runs forever. The halting problem is undecidable

in the general case, and naturally understanding the behaviour of a computer network is at least as hard as understanding the behaviour of one computer.

However, there are many interesting special cases that are decidable. In particular, it is possible to reason about the behaviour of a network of finite-state machines. One example is telling whether a given network of interacting (asynchronous and non-deterministic) finite-state machines can reach a deadlock. This problem is PSPACE-complete,[38] i.e., it is decidable, but it is not likely that there is an efficient (centralised, parallel or distributed) algorithm that solves the problem in the case of large networks.

Architectures

Various hardware and software architectures are used for distributed computing. At a lower level, it is necessary to interconnect multiple CPUs with some sort of network, regardless of whether that network is printed onto a circuit board or made up of loosely-coupled devices and cables. At a higher level, it is necessary to interconnect processes running on those CPUs with some sort of communication system.

Distributed programming typically falls into one of several basic architectures or categories: Client-server, 3-tier architecture, N-tier architecture, Distributed objects, loose coupling, or tight coupling.

- Client-server — Smart client code contacts the server for data, then formats and displays it to the user. Input at the client is committed back to the server when it represents a permanent change.
- 3-tier architecture — Three tier systems move the client intelligence to a middle tier so that stateless clients can be used. This simplifies application deployment. Most web applications are 3-Tier.
- N-tier architecture — N-Tier refers typically to web applications which further forward their requests to other enterprise services. This type of application is the one most responsible for the success of application servers.
- Tightly coupled (clustered) — refers typically to a cluster of machines that closely work together, running a shared process in parallel. The task is subdivided in parts that are made individually by each one and then put back together to make the final result.
- Peer-to-peer — an architecture where there is no special machine or machines that provide a service or manage the network resources. Instead all responsibilities are uniformly divided among all machines, known as peers. Peers can serve both as clients and servers.
- Space based — refers to an infrastructure that creates the illusion (virtualization) of one single address-space. Data are transparently replicated according to application needs. Decoupling in time, space and reference is achieved.

Another basic aspect of distributed computing architecture is the method of communicating and coordinating work among concurrent processes. Through various message passing protocols, processes may communicate directly with one another, typically in a master/slave relationship. Alternatively, a "database-centric" architecture can enable distributed computing to be done without any form of direct inter-process communication, by utilizing a shared database.[39]

See also

- List of important publications in concurrent, parallel, and distributed computing
- Edsger W. Dijkstra Prize in Distributed Computing
- List of distributed computing conferences
- List of distributed computing projects

Notes

1. ^ Andrews (2000). Dolev (2000). Ghosh (2007), p. 10.
2. ^ Godfrey (2002).
3. ^ Lynch (1996), p. 1.
4. ^ Andrews (2000), p. 291–292. Dolev (2000), p. 5.
5. ^ Ghosh (2007), p. 10.
6. ^ Andrews (2000), p. 8–9, 291. Dolev (2000), p. 5. Ghosh (2007), p. 3. Lynch (1996), p. xix, 1. Peleg (2000), p. xv.
7. ^ Andrews (2000), p. 291. Ghosh (2007), p. 3. Peleg (2000), p. 4.
8. ^ Ghosh (2007), p. 3–4. Peleg (2000), p. 1.
9. ^ Ghosh (2007), p. 4. Peleg (2000), p. 2.
10. ^ Ghosh (2007), p. 4, 8. Lynch (1996), p. 2–3. Peleg (2000), p. 4.
11. ^ Lynch (1996), p. 2. Peleg (2000), p. 1.
12. ^ Ghosh (2007), p. 7. Lynch (1996), p. xix, 2. Peleg (2000), p. 4.
13. ^ Ghosh (2007), p. 10. Keidar (2008).
14. ^ Lynch (1996), p. xix, 1–2. Peleg (2000), p. 1.
15. ^ Peleg (2000), p. 1.
16. ^ Ghosh (2007), p. 10.
17. ^ Papadimitriou (1994), Chapter 15. Keidar (2008).
18. ^ See references in Introduction.
19. ^ Andrews (2000), p. 348.
20. ^ Andrews (2000), p. 32.
21. ^ Peter (2004), The history of email (http://www.nethistory.info/History%20of%20the%20Internet/email.html).
22. ^ Elmasri & Navathe (2000), Section 24.1.2.
23. ^ Andrews (2000), p. 10–11. Ghosh (2007), p. 4–6. Lynch (1996), p. xix, 1. Peleg (2000), p. xv. Elmasri & Navathe (2000), Section 24.
24. ^ Cormen, Leiserson & Rivest (1990), Section 30.
25. ^ Cormen, Leiserson & Rivest (1990), Sections 28 and 29.
26. ^ Cole & Vishkin (1986). Cormen, Leiserson & Rivest (1990), Section 30.5.
27. ^ Andrews (2000), p. ix.
28. ^ Arora & Barak (2009), Section 6.7. Papadimitriou (1994), Section 15.3.
29. ^ Papadimitriou (1994), Section 15.2.
30. ^ Lynch (1996), p. 17–23.
31. ^ Peleg (2000), Sections 2.3 and 7. Linial (1992). Naor & Stockmeyer (1995).
32. ^ Lynch (1996), Sections 5–7. Ghosh (2007), Chapter 13.
33. ^ Lynch (1996), p. 99–102. Ghosh (2007), p. 192–193.
34. ^ Dolev (2000). Ghosh (2007), Chapter 17.
35. ^ Lynch (1996), Section 16. Peleg (2000), Section 6.
36. ^ Lynch (1996), Section 18. Ghosh (2007), Sections 6.2–6.3.
37. ^ Ghosh (2007), Section 6.4.
38. ^ Papadimitriou (1994), Section 19.3.
39. ^ A database-centric virtual chemistry system, J Chem Inf Model. 2006 May-Jun;46(3):1034-9 (http://www.ncbi.nlm.nih.gov/sites/entrez?db=pubmed&list_uids=16711722&cmd=Retrieve)

References

Books

- Andrews, Gregory R. (2000), *Foundations of Multithreaded, Parallel, and Distributed Programming*, Addison–Wesley, ISBN 0-201-35752-6.
- Arora, Sanjeev; Barak, Boaz (2009), *Computational Complexity – A Modern Approach*, Cambridge, ISBN 978-0-521-42426-4.
- Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L. (1990), *Introduction to Algorithms* (1st ed.), MIT Press, ISBN 0-262-03141-8.

- Dolev, Shlomi (2000), *Self-Stabilization*, MIT Press, ISBN 0-262-04178-2.
- Elmasri, Ramez; Navathe, Shamkant B. (2000), *Fundamentals of Database Systems* (3rd ed.), Addison–Wesley, ISBN 0-201-54263-3.
- Ghosh, Sukumar (2007), *Distributed Systems – An Algorithmic Approach*, Chapman & Hall/CRC, ISBN 978-1-58488-564-1.
- Lynch, Nancy A. (1996), *Distributed Algorithms*, Morgan Kaufmann, ISBN 1-55860-348-4.
- Papadimitriou, Christos H. (1994), *Computational Complexity*, Addison–Wesley, ISBN 0-201-53082-1.
- Peleg, David (2000), *Distributed Computing: A Locality-Sensitive Approach (http://www.ec-securehost.com/SIAM/DT05.html)*, SIAM, ISBN 0-89871-464-8, http://www.ec-securehost.com/SIAM/DT05.html.

Articles

- Cole, Richard; Vishkin, Uzi (1986), "Deterministic coin tossing with applications to optimal parallel list ranking", *Information and Control* **70** (1): 32–53, doi:10.1016/S0019-9958(86)80023-7 (http://dx.doi.org/10.1016/S0019-9958(86)80023-7).
- Keidar, Idit (2008), "Distributed computing column 32 – The year in review (http://webee.technion.ac.il/~idish/sigactNews/#column%2032)", *ACM SIGACT News* **39** (4): 53–54, http://webee.technion.ac.il/~idish/sigactNews/#column%2032.
- Linial, Nathan (1992), "Locality in distributed graph algorithms", *SIAM Journal on Computing* **21** (1): 193–201, doi:10.1137/0221015 (http://dx.doi.org/10.1137/0221015).
- Naor, Moni; Stockmeyer, Larry (1995), "What can be computed locally?", *SIAM Journal on Computing* **24** (6): 1259–1277, doi:10.1137/S0097539793254571 (http://dx.doi.org/10.1137/S0097539793254571).

Web sites

- Godfrey, Bill (2002). "A primer on distributed computing (http://www.bacchae.co.uk/docs/dist.html)". http://www.bacchae.co.uk/docs/dist.html.
- Peter, Ian (2004). "Ian Peter's History of the Internet (http://www.nethistory.info/History%20of%20the%20Internet/)". http://www.nethistory.info/History%20of%20the%20Internet/. Retrieved 2009-08-04.

Further reading

Books

- Tel, Gerard (1994). *Introduction to Distributed Algorithms*. Cambridge University Press.
- Attiya, Hagit and Welch, Jennifer (2004). *Distributed Computing: Fundamentals, Simulations, and Advanced Topics*. Wiley-Interscience. ISBN 0471453242.

Articles

- Keidar, Idit; Rajsbaum, Sergio, eds. (2000–2009), "Distributed computing column (http://webee.technion.ac.il/~idish/sigactNews/)", *ACM SIGACT News*, http://webee.technion.ac.il/~idish/sigactNews/.

External links

- Distributed computing (http://www.dmoz.org/Computers/Computer_Science/Distributed_Computing//) at the Open Directory Project
- Distributed computing journals (http://www.dmoz.org/Computers/Computer_Science/Distributed_Computing/Publications//) at the Open

Directory Project

Retrieved from "http://en.wikipedia.org/wiki/Distributed_computing"
Categories: Distributed computing

Building a Bastion Host Using HP-UX 11

- Launched: **Oct 16, 2002**
- Updated: **Oct 16, 2002**
- Section: Network Security Library :: Unix Security
- Author: Kevin Steves
- Company: **Hewlett-Packard Consulting**
- Rating: **4.5/5 - 82 Votes**

Very good paper. Almost FAQ.

> *This is an update to a paper I originally wrote in 1997 titled "Building a Bastion Host Using HP-UX 10". It has been modified to reflect changes in HP-UX 11, in addition to incorporating the changes in my methodology that have occurred over the last 3 years.*

A bastion host is a computer system that is exposed to attack, and may be a critical component in a network security system. Special attention must be paid to these highly fortified hosts, both during initial construction and ongoing operation. Bastion hosts can include:

- Firewall gateways
- Web servers
- FTP servers
- Name servers (DNS)
- Mail hubs
- Victim hosts (sacrificial lambs)

This paper presents a methodology for building a bastion host using HP-UX 11, and walks through the steps used to build a sample, generic bastion host using HP-UX 11.00. While the principles and procedures can be applied to other HP-UX versions as well as other Unix variants, our focus is on HP-UX 11.

What is a Bastion Host?

The American Heritage Dictionary defines a bastion as:

> 1. A projecting part of a rampart or other fortification. 2. A well-fortified position or area. 3. Something regarded as a defensive stronghold.

Marcus Ranum is generally credited with applying the term *bastion* to hosts that are exposed to attack, and its common use in the firewall community. In [1] he says:

> Bastions are the highly fortified parts of a medieval castle; points that overlook critical areas of defense, usually having stronger walls, room for extra troops, and the occasional useful tub of boiling hot oil for discouraging attackers. A bastion host is a system identified by the firewall administrator as a critical strong point in the network's security. Generally, bastion hosts will have some degree of extra attention paid to their security, may undergo regular audits, and may have modified software.

Bastion hosts are not general purpose computing resources. They differ in both their purpose and their specific configuration. A victim host may permit network logins so users can run untrusted services, while a firewall gateway may only permit logins at the system

console. The process of configuring or constructing a bastion host is often referred to as *hardening*.

The effectiveness of a specific bastion host configuration can usually be judged by answering the following questions:

1. How does the bastion host protect *itself* from attack?
2. How does the bastion host protect *the network behind it* from attack?

Extreme caution should be exercised when installing new software on bastion hosts. Very few software products have been designed and tested to run on these exposed systems.

See [2] for a thorough treatment of bastion hosts.

Methodology

Let's begin by creating a *methodology*. These are the principles and procedures we will follow as we build bastion hosts. Included in this is our *mindset*, which will help guide the configuration decisions we make.

We take a paranoid stance--what we don't know can hurt us, and what we think we know we may not trust. We start with a clean operating system install. If subsystems are not *needed* for the applications we plan to run on the bastion host, we will not install them in the first place, or disable or remove them after the install. Next we install any additional operating system software needed on the bastion host, such as network drivers not available on the install media or the LVM Mirror product, followed by the latest patch bundle (Support Plus Bundle). We perform a security patch review and install HP-UX security patches that apply to our installed software configuration. The system is configured with commercial security (as a trusted system) which removes the hashed passwords from the `/etc/passwd` file and provides other useful security features such as auditing and login passwords with lengths greater than 8 characters. Unneeded pseudo-accounts in the password database are removed. We remove the set-id bits from all programs then selectively add them back to programs that must be run by non-privileged users. This proactive approach may save us time and a future vulnerability window when the next security defect is discovered in a set-id program. We tighten up the world-write permissions on system files, and set the sticky bit on publicly writable directories. We next set a number of tunable network parameters with a paranoid stance toward security. At this point, the applications that will run on the bastion host can be installed, configured and tested. This may include installing additional security software, such as TCP wrappers and SSH. After testing is complete, we create a bootable System Recovery Tape of the root volume group.

Sample Blueprint

Now let's lay out the blueprint that we'll use as we construct a sample, generic bastion host using HP-UX 11.00:

1. Install HP-UX
2. Install Additional Products
3. Install Support Plus Bundle
4. Install Security Patches
5. First Steps
6. Disable Network Services
7. Disable Other Daemons
8. Examine Set-id Programs
9. Examine File Permissions
10. Security Network Tuning
11. Install Software and Test Configuration
12. Create System Recovery Tape

Keep in mind that this is a *sample* starting configuration, and you will need to make changes specific to your planned use of the system. If you're installing a future HP-UX version like 11.10, some things may be different. You may also choose to reorder things slightly for various reasons. Every bastion host that I have configured has been different. Document your configuration steps as you perform them--you may discover later that a change that was made causes unforseen problems. And it may take several install iterations to get everything working correctly.

1. Install HP-UX

It takes at most one hour to install a minimal HP-UX configuration from CD-ROM. The security benefits of starting with a clean operating system install, and knowing exactly what you have, far exceed this minor cost in your time. Even if your host is new and has been shipped from the factory with HP-UX preinstalled, you should reinstall from scratch.

During the initial installation, configuration and testing, make sure that your system is not connected to any untrusted networks. You may want to only connect the system to a network *after* you have completed your configuration steps. In this example I used a completely private network (e.g., hub or cross-cable) connected only to the LAN console.

Note the test system used is an L2000, which will only run 64-bit HP-UX; we are also using the 9911 install media (11.ACE).

To perform the installation we boot from the install CD and perform the following steps:

1. Select "Install HP-UX"
2. In the "User Interface and Media Options" screen select:
    1. Media only installation
    2. Advanced Installation
3. In the "Basic" screen select Environments "64-Bit Minimal HP-UX (English Only)"
4. In the "Software" screen:
    1. Select "Change Depot Location"
    2. Change "Interactive swinstall" to "Yes"
    3. Select "Modify"
5. Change other configuration settings as appropriate for your system
6. Select "Go!"
7. In the "SD Install" screen:
    1. Change the Software View to Products:
       View->Change Software View->Start with Products
    2. Mark MailUtilities.Runtime and MailUtilities.Manuals for Install
    3. Unmark NFS.Runtime.NIS-CLIENT for Install (this will also unmark KEY-CORE and NIS-CORE)
    4. Unmark NFS.Runtime.NFS-CLIENT for Install
    5. Mark NFS.Runtime.NFS-64SLIB for Install
    6. Unmark Networking.MinimumRuntime.PPP-RUN for Install
    7. Select OS-Core.Manuals for Install
    8. Select SOE for Install
    9. Select SecurityMon for Install
    10. Select Streams.Runtime.STREAMS-64SLIB for Install
    11. Select SystemAdmin.Runtime for Install
    12. Select TextEditors.Runtime and TextEditors.Manuals for Install
    13. Perform installation analysis:
        Actions->Install (analysis)

We choose a minimal HP-UX system. This will not install the X window system and many other products that we don't need or want. We remove as much of the NFS product as possible because it has a number of security problems and we will not be using it. We also remove the PPP-RUN fileset because we are not using PPP. For system management purposes we install SAM, the core OS man pages, mailers and text editors. We will be using the commercial security feature of HP-UX so we need to select the SecurityMon and SOE products. Finally, since we are installing on 64-bit hardware, we select the 64-bit libraries for NFS and STREAMS which are required for various applications.

We would like to remove other products such as SNMP (OVSNMPAgent) but a number of other products are dependent upon it (which seems questionable). We will disable SNMP and other products that are difficult or impossible to remove.

This yields a relatively lean configuration (much of the space in `/var/` is for saved patches which we can optionally remove later) as shown by the following output of `bdf`, `ps -ef` and `netstat -anf inet` (but we still have work to do):

```
# uname -a
HP-UX bastion B.11.00 A 9000/800 137901517 two-user license

# bdf
Filesystem          kbytes    used   avail %used Mounted on
/dev/vg00/lvol3     143360   18699  116899   14% /
/dev/vg00/lvol1      83733   15965   59394   21% /stand
```

```
/dev/vg00/lvol8     512000  123680  364879   25% /var
/dev/vg00/lvol7     512000  164352  325949   34% /usr
/dev/vg00/lvol4      65536    1122   60394    2% /tmp
/dev/vg00/lvol6     262144    3513  242523    1% /opt
/dev/vg00/lvol5      20480    1109   18168    6% /home

# ps -ef
     UID   PID  PPID  C    STIME TTY       TIME COMMAND
    root     0     0  0 14:21:25 ?         0:10 swapper
    root     1     0  0 14:21:25 ?         0:00 init
    root     2     0  0 14:21:25 ?         0:00 vhand
    root     3     0  0 14:21:25 ?         0:00 statdaemon
    root     4     0  0 14:21:25 ?         0:00 unhashdaemon
    root     8     0  0 14:21:25 ?         0:00 supsched
    root     9     0  0 14:21:25 ?         0:00 strmem
    root    10     0  0 14:21:25 ?         0:00 strweld
    root    11     0  0 14:21:25 ?         0:00 strfreebd
    root    12     0  0 14:21:25 ?         0:00 ttisr
    root    18     0  0 14:21:25 ?         0:00 lvmkd
    root    19     0  0 14:21:25 ?         0:00 lvmkd
    root    20     0  0 14:21:25 ?         0:00 lvmkd
    root    21     0  0 14:21:25 ?         0:00 lvmkd
    root    22     0  0 14:21:25 ?         0:00 lvmkd
    root    23     0  0 14:21:25 ?         0:00 lvmkd
    root   826     1  0 14:25:12 console   0:00 -sh
    root   522     1  0 14:24:48 ?         0:00 /usr/sbin/ptydaemon
    root   870   866  1 14:30:26 console   0:00 ps -ef
    root    28     0  0 14:21:26 ?         0:00 vxfsd
    root   460     1  0 14:24:46 ?         0:00 /usr/sbin/syncer
    root   708     1  0 14:24:58 ?         0:00 /usr/sbin/snmpdm
    root   651     1  0 14:24:57 ?         0:00 /usr/sbin/rpcbind
    root   519     1  0 14:24:48 ?         0:00 /usr/sbin/syslogd -D
    root   535     1  0 14:24:49 ?         0:00 /usr/lbin/nktl_daemon 0 0 0 0 0 1 -2
    root   656     0  0 14:24:57 ?         0:00 nfskd
    root   545     1  0 14:24:52 ?         0:00 /usr/lbin/ntl_reader 0 1 1 1 1000 /var/adm
    root   546   545  0 14:24:52 ?         0:00 /usr/sbin/netfmt -C -F -f /var/adm/nettl.L
    root   746     1  0 14:25:09 ?         0:00 /usr/sbin/cron
    root   680     1  0 14:24:57 ?         0:00 /usr/sbin/inetd
    root   703     1  0 14:24:58 ?         0:00 sendmail: accepting connections on port 25
    root   866   826  0 14:28:53 console   0:00 ksh
    root   719     1  0 14:25:08 ?         0:00 /usr/sbin/hp_unixagt
    root   727     1  0 14:25:09 ?         0:06 /usr/sbin/mib2agt
    root   735     1  0 14:25:09 ?         0:00 /usr/sbin/trapdestagt
    root   743     1  0 14:25:09 ?         0:00 /usr/sbin/pwgrd
    root   749     1  0 14:25:09 ?         0:00 /usr/sbin/envd
    root   758     1  0 14:25:09 ?         0:00 /usr/sbin/swagentd -r

# netstat -anf inet
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address          Foreign Address        (state)
tcp        0      0  *.7161                 *.*                     LISTEN
tcp        0      0  *.544                  *.*                     LISTEN
tcp        0      0  *.543                  *.*                     LISTEN
tcp        0      0  *.515                  *.*                     LISTEN
tcp        0      0  *.514                  *.*                     LISTEN
tcp        0      0  *.513                  *.*                     LISTEN
tcp        0      0  *.512                  *.*                     LISTEN
tcp        0      0  *.113                  *.*                     LISTEN
tcp        0      0  *.111                  *.*                     LISTEN
tcp        0      0  *.37                   *.*                     LISTEN
tcp        0      0  *.25                   *.*                     LISTEN
tcp        0      0  *.23                   *.*                     LISTEN
```

```
tcp        0      0  *.21                  *.*                        LISTEN
tcp        0      0  *.19                  *.*                        LISTEN
tcp        0      0  *.13                  *.*                        LISTEN
tcp        0      0  *.9                   *.*                        LISTEN
tcp        0      0  *.7                   *.*                        LISTEN
udp        0      0  *.2121                *.*
udp        0      0  *.514                 *.*
udp        0      0  *.111                 *.*
udp        0      0  *.*                   *.*
udp        0      0  *.49152               *.*
udp        0      0  *.518                 *.*
udp        0      0  *.13                  *.*
udp        0      0  *.7                   *.*
udp        0      0  *.9                   *.*
udp        0      0  *.19                  *.*
udp        0      0  *.161                 *.*
udp        0      0  *.*                   *.*
udp        0      0  *.*                   *.*
udp        0      0  *.*                   *.*
```

2. Install Additional Products

At this point, you should install any additional HP products that are required on the bastion host, for example network drivers for add-on LAN cards, or other products you plan to use like LVM Mirror. You will want to install a portion of the HP Ignite product to obtain the software (`make_recovery` command) required to build a bootable backup tape of the root volume group, which we will create at the end of the configuration process.

For our sample configuration, we are using the 4-Port 100BT PCI card, so we need to install the driver for that card, and we will also install the required filesets in Ignite-UX for `make_recovery` functionality.

Using the December 1999 Applications CD we install the following product and filesets:

1. 100BASE-T
2. Ignite-UX.BOOT-KERNEL
3. Ignite-UX.FILE-SRV-11-00
4. Ignite-UX.MGMT-TOOLS
5. Ignite-UX.RECOVERY

3. Install Support Plus Bundle

Next we install all General Release (GR) patches from the latest HP-UX 11.0 Support Plus CD, which in the example is from December 1999. The install CD contained a recent set of patches from around when the media was produced, which was November 1999, so we don't expect to have many patches that are selected. Mount the Support Plus CD and use `swinstall` to install the GR bundle XSWGR1100.

4. Install Security Patches

We next perform a security patch review, to determine if any security patches should be installed. HP-UX patches are available via anonymous FTP [3]. An "HP-UX Patch Security Matrix" [4] is also available, which contains a list of current security patches for each HP-UX platform and operating system version combination (e.g., s800 11.00). The matrix is updated nightly. There is also a list of the MD5 hash codes [5] for each patch which can be used to verify that patches you intend to install have not been tampered with (though it would be nice if this file was in turn PGP signed).

For our sample s800, 11.00 host, at the time of this writing, the current security patches are:

```
s800 11.00:PHCO_19945 s700_800 11.00 bdf(1M) patch to skip autofs file systems
           PHCO_20078 s700_800 11.0 Software Distributor (SD-UX) Cumulative Patch
           PHCO_20765 s700_800 11.00 libc cumulative patch
```

```
PHKL_20315 s700_800 11.00 Cumulative LOFS patch
PHNE_16295 s700_800 11.00 vacation patch.
PHNE_17028 s700_800 11.00 r-commands cumulative mega-patch
PHNE_17190 s700_800 11.00 sendmail(1m) 8.8.6 patch
PHNE_17949 s700_800 11.00 Domain Management (DESMS B.01.12)
PHNE_18017 s700_800 11.00 Domain Management (DESMS-NS B.01.11)
PHNE_18377 s700_800 11.00 ftpd(1M) and ftp(1) patch
PHNE_19620 s700_800 11.0 ONC cumulative patch
PHNE_20619 s700_800 11.00 Bind 4.9.7 components
PHNE_20735 s700_800 11.00 cumulative ARPA Transport patch
PHSS_16649 s700_800 11.00 Receiver Services October 1998 Patch
PHSS_17310 s700_800 11.00 OV OB2.55 patch - WinNT packet
PHSS_17483 s700_800 11.00 MC/LockManager A.11.05 (English) Patch
PHSS_17484 s700_800 11.00 MC/LockManager A.11.05 (Japanese) Patch
PHSS_17496 s700_800 11.00 Predictive C.11.0[0,a-m] cumulative patch
PHSS_17581 s700_800 11.00 MC ServiceGuard 11.05 Cumulative Patch
PHSS_20385 s700_800 11.00 OV OB2.55 patch - DA packet
PHSS_20544 s700_800 11.00 OV EMANATE14.2 Agent Consolidated Patch
PHSS_20716 s700_800 11.00 CDE Runtime DEC99 Periodic Patch
```

Each patch for a product currently installed on the system should be analyzed to determine if it needs to be installed. First you should check and see if it's already installed from either the install media or the patch bundle. If not, you can look at the the patch .text file for details about the patch, including dependencies, filesets effected, and files patched. You can determine filesets installed on the system by executing `swlist -l fileset`.

Just because a patch exists doesn't mean that you need to install it, though it is safest to do so. Some patches may fix buffer overrun defects or other attack channels in set-uid root commands or root processes. If you plan to remove the set-uid bits you may choose not to install them. You may also not have a program configured (for example, `rlogind` listening on the network), but sometimes it can be difficult to determine if a defect is remotely or locally exploitable. If you're not sure whether a particular patch needs to be installed, it's best to just install it.

You should also examine the security bulletins themselves [6], because not all security bulletins result in a patch, for example there is a security bulletin regarding the default PMTU strategy that recommends its default be changed using `ndd` (HPSBUX0001-110) and also a serious issue with blank password fields when using Ignite-UX and trusted systems (HPSBUX0002-111). We will address the issue with the PMTU setting below when we set network security tunables, and the Ignite-UX issue concerns `make_sys_image`, which we will not be using.

5. First Steps

There are a few, miscellaneous configuration and cleanup steps we can perform immediately after the operating system install and patch steps.

1. Optionally remove saved patches.

   By default during patch installation, rollback copies of all patch files modified are saved in `/var/adm/sw/save/`. You may wish to remove these files and claim the disk space by marking the patches "committed". However, if you do this, there will be no way to uninstall the patch with `swremove`. I tend to remove saved patches following a fresh install. To do this perform the following:

   ```
   # swmodify -x patch_commit=true '*.*'
   ```

2. Convert to a trusted system.

   ```
   # /usr/lbin/tsconvert
   Creating secure password database...
   Directories created.
   Making default files.
   System default file created...
   Terminal default file created...
   ```

```
Device assignment file created...
Moving passwords...
secure password database installed.
Converting at and crontab jobs...
At and crontab files converted.
# passwd root
```

Passwords on existing accounts will expire as a result of the conversion, which is why we change the root password.

You may also consider enabling auditing.

3. Tighten global privileges.

HP-UX has a feature known as privilege groups, which is mechanism to assign a privilege to a group (see `privgrp(4)`). By default the `CHOWN` privilege is a global privilege and applies to all groups:

```
$ getprivgrp
global privileges: CHOWN
```

Non-privileged users really don't need to be able to chown files to other users; in Linux for example, only the super-user may change the owner of a file. `/sbin/init.d/set_prvgrp` is executed by default at system startup and executes the command `/usr/sbin/setprivgrp -f /etc/privgroup` if `/etc/privgroup` exists. We can create a configuration file that will delete all privileges for all groups (see `setprivgrp(1m)`):

```
# getprivgrp
global privileges: CHOWN
# echo -n >/etc/privgroup
# chmod 400 /etc/privgroup
# /sbin/init.d/set_prvgrp start
# getprivgrp
global privileges:
```

4. Fix PAM CDE problems.

SAM will perform some correctness checks on `/etc/pam.conf` that involve trying to find a command using several different paths for each `service_name`. We did not install CDE and yet our `pam.conf` file contains `dtlogin` and `dtaction` entries for each of the PAM module types; for example:

```
dtlogin  auth required  /usr/lib/security/libpam_unix.1
dtaction auth required  /usr/lib/security/libpam_unix.1
```

We can safely remove these, which will permit us to access the authenticated commands functionality in SAM:

```
# cp /etc/pam.conf /etc/pam.conf.SAVE
# grep -Ev '^(dtlogin|dtaction)' /etc/pam.conf.SAVE >/etc/pam.conf
```

5. Fix hparray startup weirdness.

For some reason there are some startup symlinks pointing to array startup scripts that are contained in filesets that we do not have and do not need (`OS-Core.C2400-UTIL` and `OS-Core.ARRAY-MGMT`) so we remove them:

```
# for f in /sbin/rc*.d/*; do [ ! -f $f ] && echo $f; done
/sbin/rc1.d/K290hparamgr
/sbin/rc1.d/K290hparray
/sbin/rc2.d/S710hparamgr
/sbin/rc2.d/S710hparray
# rm /sbin/rc1.d/K290hparamgr
```

```
# rm /sbin/rc1.d/K290hparray
# rm /sbin/rc2.d/S710hparamgr
# rm /sbin/rc2.d/S710hparray
```

6. Set default umask.

   One side-effect of converting to a trusted system, is the default umask of 0 is changed to 07077, so nothing needs to be performed to tighten up the umask.

7. Restrict root login to the console if desired.

```
# echo console > /etc/securetty
# chmod 400 /etc/securetty
```

8. Enable inetd logging if inetd will remain enabled.

   Add the -l (minus ell) argument to the INETD_ARGS environment variable in `/etc/rc.config.d/netdaemons`:

```
export INETD_ARGS=-l
```

9. Remove unneeded pseudo-accounts.

   First we examine some groups that might be removed, then users; our basic strategy is if there are no processes that are run with a given user or group, and there are no files owned by a user or group, we remove them:

```
# find / -group lp -o -group nuucp daemon -exec ls -ld {} \;
# groupdel lp
# groupdel nuucp
# groupdel daemon
# find / -user uucp -o -user lp -o -user nuucp -o -user hpdb \
> -o -user www -o -user daemon -exec ls -ld {} \;
# userdel uucp
# userdel lp
# userdel nuucp
# userdel hpdb
# userdel www
# userdel daemon
```

   For the remaining pseudo-accounts (`bin`, `sys` and `adm`), you should change the login shell to some invalid path, for example `/`, or consider using the `noshell` program from the Titan package [7].

```
# pwget -n bin
bin:*:2:2:NO LOGIN:/usr/bin:/
```

10. Configure `nsswitch.conf(4)` policy.

    If you are going to configure the DNS resolver you can do it at this point. Many bastion hosts, including firewall gateways, do not have DNS configured at all. For these hosts, you can set the `nsswitch.conf(4)` to search local files only:

```
# cp /etc/nsswitch.files /etc/nsswitch.conf
# chmod 444 /etc/nsswitch.conf
```

11. Change `root` home directory to `/root`.

    We change root's home directory from the default of `/` to `/root`. Our motivation is to give the `root` account a private home directory to lessen the possibility of files being placed unintentionally in `/`, and it also permits us to put a restrictive mode on the directory. Edit `/etc/passwd` and change root's entry to:

```
root:*:0:3::/root:/sbin/sh
```

Then build the directory and update the TCB:

```
# mkdir /root
# chmod 700 /root
# mv /.profile /root
# pwconv
Updating the tcb to match /etc/passwd, if needed.
```

6. Disable Network Services

Disable inetd Services

We should be able to identify each TCP and UDP service emitted by `netstat -af inet`. Those that are not needed or cannot be secured should be disabled. Examples of such services include the UDP and TCP small servers, like echo, chargen, daytime, time and discard; the Berkeley r* services, talk, etc. Some bastion hosts have an *entirely empty* `inetd.conf`. We can start by removing all services from `inetd.conf`, restarting it, then examining the `netstat` output. If you stick with a bare `inetd.conf`, you can choose to not run `inetd` at all. You can disable `inetd` startup and shutdown by removing the corresponding symbolic links from the rc directories:

```
# rm /sbin/rc2.d/S500inetd
# rm /sbin/rc1.d/K500inetd
```

For the remaining services, consider using `inetd.sec(4)`, which permits IP address based authentication of remote systems.

With all services removed from `inetd.conf`, `netstat` yields:

```
# netstat -af inet
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address          Foreign Address        (state)
tcp        0      0  *.7161                 *.*                     LISTEN
tcp        0      0  *.portmap              *.*                     LISTEN
tcp        0      0  *.smtp                 *.*                     LISTEN
udp        0      0  *.2121                 *.*
udp        0      0  *.syslog               *.*
udp        0      0  *.portmap              *.*
udp        0      0  *.*                    *.*
udp        0      0  *.49152                *.*
udp        0      0  *.*                    *.*
udp        0      0  *.snmp                 *.*
udp        0      0  *.*                    *.*
udp        0      0  *.*                    *.*
```

This is much better, though we still need to determine what the remaining services are. We see that servers are listening on the UDP SNMP, portmap and syslog ports, and the SMTP and TCP portmap ports. However, 2121/udp, 2121/tcp, 7161/tcp and 49152/udp were not found in `/etc/services`, so `netstat` is unable to print the service name. There are also some wildcard (*.*) local UDP listeners that are a mystery.

An *extremely* useful tool for identifying network services is `lsof` (LiSt Open Files) [8]. `lsof -i` shows us the processes that are listening on the remaining ports:

```
# lsof -i
COMMAND   PID USER   FD   TYPE      DEVICE SIZE/OFF NODE NAME
syslogd   261 root   5u   inet 0x10191e868      0t0  UDP *:syslog (Idle)
rpcbind   345 root   4u   inet     72,0x73      0t0  UDP *:portmap (Idle)
rpcbind   345 root   6u   inet     72,0x73      0t0  UDP *:49158 (Idle)
rpcbind   345 root   7u   inet     72,0x72      0t0  TCP *:portmap (LISTEN)
```

```
sendmail: 397 root    5u  inet 0x10222b668      0t0  TCP *:smtp (LISTEN)
snmpdm    402 root    3u  inet 0x10221a268      0t0  TCP *:7161 (LISTEN)
snmpdm    402 root    5u  inet 0x10222a268      0t0  UDP *:snmp (Idle)
snmpdm    402 root    6u  inet 0x10221f868      0t0  UDP *:* (Unbound)
mib2agt   421 root    0u  inet 0x10223e868      0t0  UDP *:* (Unbound)
swagentd  453 root    6u  inet 0x1019d3268      0t0  UDP *:2121 (Idle)
```

We see that `rpcbind` is listening on 49158/udp (it's unclear whether this is a fixed or ephemeral port assignment) and `snmpdm` is listening on 7161/tcp. Also, we see that `snmpdm` and `mib2agt` are the source of the mysterious unbound wildcard ports.

Disable Other Services

With this information, we can proceed with the following steps.

1. Prevent `syslogd` from listening on the network.

   PHCO_21023 can be installed which adds a `-N` option to `syslogd` to prevent it from listening on the network for remote log messages. After installing this patch, edit `/sbin/init.d/syslogd` and modify the line that starts `syslogd` to be `/usr/sbin/syslogd -DN`.

2. Disable SNMP daemons.

   Edit SNMP startup configuration files:
   1. `/etc/rc.config.d/SnmpHpunix`
      Set SNMP_HPUNIX_START to 0: SNMP_HPUNIX_START=0
   2. `/etc/rc.config.d/SnmpMaster`
      Set SNMP_MASTER_START to 0: SNMP_MASTER_START=0
   3. `/etc/rc.config.d/SnmpMib2`
      Set SNMP_MIB2_START to 0: SNMP_MIB2_START=0
   4. `/etc/rc.config.d/SnmpTrpDst`
      Set SNMP_TRAPDEST_START to 0: SNMP_TRAPDEST_START=0

3. Disable `swagentd` (SD-UX) daemon.

   This is complicated. The `swagentd` script is run twice in the bootup start sequence, and performs different tasks based upon its program name argument. For example, if run as `S100swagentd` it will remove the files listed in `/var/adm/sw/cleanupfile`. Also, for the `swconfig` script to work properly, `swagentd` must be running. Our solution is to create a new script, that will be configured to run immediately after `S120swconfig` to kill the `swagentd` daemon in a paranoid fashion, and remove the other start and kill rc links.

   The key portion of the kill script, `swagentdk` [9], follows:

```
start)
        /usr/sbin/swagentd -k
        sleep 1
        findproc swagentd
        if [ "$pid" != "" ]; then
                kill $pid
                sleep 5
                findproc swagentd
                if [ "$pid" != "" ]; then
                        kill -9 $pid
                        sleep 5
                        findproc swagentd
                        if [ "$pid" != "" ]; then
                                echo "UNABLE TO KILL SWAGENTD PROCESS!!!"
                                rval=3  # REBOOT!!!
                        fi
```

```
                else
                        rval=0
                fi
        else
                rval=0
        fi
        ;;
```

We try to kill the daemon 3 times, with increasing levels of force. If we can't stop the daemon using `kill -9`, we set `rval=3`, which will cause a reboot (this drastic step may exceed your specific security and paranoia requirements).

To configure, perform the following:

```
# cp /tmp/swagentdk /sbin/init.d
# chmod 555 /sbin/init.d/swagentdk
# ln -s /sbin/init.d/swagentdk /sbin/rc2.d/S121swagentdk
# rm /sbin/rc2.d/S870swagentd
# rm /sbin/rc1.d/K900swagentd
```

4. Disable `sendmail` daemon.

   Set the SENDMAIL_SERVER environment variable to 0 in `/etc/rc.config.d/mailservs`:

```
export SENDMAIL_SERVER=0
```

5. Disable `rpcbind` daemon.

   We don't plan to run any RPC services on the bastion host and need to disable the startup of `rpcbind` (this is the `portmap` replacement on HP-UX 11.0). After some grepping in `/etc/rc.config.d` we find that `rpcbind` is started from the `nfs.core` script, so we disable it in the rc startup directories. We also move the `rpcbind` program to a new name as an additional safety measure (though a patch install could reinstall it so it's important to reexamine your configuration after patches are installed on the bastion host):

```
# rm /sbin/rc1.d/K600nfs.core
# rm /sbin/rc2.d/S400nfs.core
# mv /usr/sbin/rpcbind /usr/sbin/rpcbind.DISABLE
```

   This also avoids the startup of the `nfskd` process, which we saw in previous `ps` output.

After a reboot to verify the modifications made to the startup scripts, we can check the `netstat` and `lsof` output and verify that no network services remain enabled. We can also check the `ps` output again to verify that the disabled daemons were not launched:

```
# netstat -af inet
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address          Foreign Address        (state)
udp        0      0  *.*
# lsof -i
# ps -ef
     UID   PID  PPID  C    STIME TTY       TIME COMMAND
    root     0     0  0 15:59:18 ?         0:10 swapper
    root     1     0  0 15:59:19 ?         0:00 init
    root     2     0  0 15:59:18 ?         0:00 vhand
    root     3     0  0 15:59:18 ?         0:00 statdaemon
    root     4     0  0 15:59:18 ?         0:00 unhashdaemon
    root     8     0  0 15:59:18 ?         0:00 supsched
    root     9     0  0 15:59:18 ?         0:00 strmem
    root    10     0  0 15:59:18 ?         0:00 strweld
    root    11     0  0 15:59:18 ?         0:00 strfreebd
```

```
    root    12     0  0 15:59:18 ?          0:00 ttisr
    root    18     0  0 15:59:19 ?          0:00 lvmkd
    root    19     0  0 15:59:19 ?          0:00 lvmkd
    root    20     0  0 15:59:19 ?          0:00 lvmkd
    root    21     0  0 15:59:19 ?          0:00 lvmkd
    root    22     0  0 15:59:19 ?          0:00 lvmkd
    root    23     0  0 15:59:19 ?          0:00 lvmkd
    root   367     1  0 15:59:48 console    0:00 -sh
    root   206     1  0 15:59:38 ?          0:00 /usr/sbin/syncer
    root   324     1  0 15:59:47 ?          0:00 /usr/sbin/inetd -l
    root    28     0  0 15:59:20 ?          0:00 vxfsd
    root   237     1  0 15:59:39 ?          0:00 /usr/sbin/ptydaemon
    root   380   367  0 16:00:03 console    0:00 ksh
    root   410   380  1 16:04:05 console    0:00 ps -ef
    root   250     1  0 15:59:40 ?          0:00 /usr/lbin/nktl_daemon 0 0 0 0 0 1 -2
    root   356     1  0 15:59:47 ?          0:00 /usr/sbin/cron
    root   260     1  0 15:59:42 ?          0:00 /usr/lbin/ntl_reader 0 1 1 1 1000 /var/adm
    root   261   260  0 15:59:42 ?          0:00 /usr/sbin/netfmt -C -F -f /var/adm/nettl.L
    root   352     1  0 15:59:47 ?          0:00 /usr/sbin/pwgrd
    root   359     1  0 15:59:47 ?          0:00 /usr/sbin/envd
    root   400     1  0 16:02:04 ?          0:00 /usr/sbin/syslogd -DN
```

For some unknown reason, `netstat` shows a wildcard UDP listener, but `lsof` is silent on this. This is a concern, and I have notified the HP-UX networking lab about this, and they are investigating.

7. Disable Other Daemons

We can now examine the current process listing and determine if there are other daemons that can be disabled. Our approach is: if we aren't using it, disable it. Many of the processes remaining are system processes. System processes can be identified by examining the flags column in a long process listing (`ps -el`); flags is an additive octal bit-field, like the Unix mode bits on files (see `ps(1)` for a listing of the process flag bits). The processes that have the 2 flag bit set (e.g. 1003, 01000 + 2 + 1) are system processes and can probably be ignored safely (the 01000 bit is explained below):

```
# ps -el
  F S      UID   PID  PPID  C PRI NI         ADDR   SZ          WCHAN TTY       TI
1003 S       0     0     0  0 128 20       6a4f58    0              - ?          0
141 S      0     1     0  0 168 20    101d3e600  100 400003fffff0000 ?          0:
1003 S       0     2     0  0 128 20    101b25f00    0        747e90 ?          0
1003 S       0     3     0  0 128 20    101b36200    0        5f2060 ?          0
1003 S       0     4     0  0 128 20    101b36500    0        6ec250 ?          0
1003 S       0     8     0  0 100 20    101b25300    0        72fed8 ?          0
1003 S       0     9     0  0 100 20    101b25600    0        6a3698 ?          0
1003 S       0    10     0  0 100 20    101b25900    0        6f2988 ?          0
1003 S       0    11     0  0 100 20    101b25c00    0        6cc2d0 ?          0
1003 S       0    12     0  0 -32 20    101b36800    0        6a0c68 ?          0
1003 S       0    18     0  0 147 20    101b4c000    0        6a2fb0 ?          0
1003 S       0    19     0  0 147 20    101b4c300    0        6a2fb0 ?          0
1003 S       0    20     0  0 147 20    101b4c600    0        6a2fb0 ?          0
1003 S       0    21     0  0 147 20    101b4c900    0        6a2fb0 ?          0
1003 S       0    22     0  0 147 20    101b4cc00    0        6a2fb0 ?          0
1003 S       0    23     0  0 147 20    101b4cf00    0        6a2fb0 ?          0
  1 S      0   367     1  0 158 20    101e56100  106       31fff00 console   0:
  1 S      0   206     1  0 154 20    101df9b00    7        6a201c ?          0:
  1 S      0   324     1  0 168 20    1019f0d00   24 400003fffff0000 ?          0:
1003 R       0    28     0  0 152 20    101b7a900    0              - ?          0
  1 S      0   237     1  0 155 20    1019cb600   20        701ef0 ?          0:
  1 S      0   380   367  0 158 20    101b60500   48       32011c0 console   0:
  1 S      0   250     1  0 127 20    1019f6d00   15        623a74 ?          0:
  1 S      0   356     1  0 154 20    101e56800   19      101b76d2e ?          0:
  1 S      0   260     1  0 127 20    1019a5200   18        6f2e8c ?          0:
```

```
1 S       0   261   260  0 127 20    1019f8b00  29    1019f75c0 ?        0:
1 S       0   352     1  0 154 20    101e3d500  46       746ca4 ?        0:
1 S       0   359     1  0 154 20    101e5db00  14    1019a652e ?        0:
1 S       0   400     1  0 154 20    1019a7f00  21       746ca4 ?        0:
1 R       0   413   380  0 157 20    1019a7400  25            - console  0:
```

Not all flag bits are documented in `ps(1)`; undocumented flag bits include:

- 040 - process' text locked in memory
- 0100 - process' data locked in memory
- 0200 - enables per-process syscall tracing
- 0400 - process has one or more lazy swap regions
- 01000 - process has 64-bit address space

This explains the 141 value seen for `init`: it has 0100 set because data is locked in memory, 040 because the text is locked in memory, and 1 because it's currently in core (0100 + 040 + 1 = 141), and the 1003 value for system processes like `lvmkd` (01000 + 2 + 1) which in this example, are 64-bit.

The list of non-system processes include:

- `init`
- `syncer`
- `inetd`
- `ptydaemon`
- `nktl_daemon, ntl_reader, netfmt`
- `cron`
- `pwgrd`
- `envd`
- `syslogd`

By examining the man pages available for these daemons we determine that we need most of them. As mentioned earlier, you can disable `inetd` if you have no inetd-launched services. I suppose `cron` could be disabled if you do not plan to have any cron jobs, but that seems unlikely.

`envd` logs messages and can perform actions when over-temperature and chassis fan failure conditions are detected by the hardware. For example, in its default configuration it will execute `/usr/sbin/reboot -qh` when the temperature has exceeded the maximum operating limit of the hardware, in an attempt to preserve data integrity. I leave this daemon running, but you can disable its startup by modifying `/etc/rc.config.d/envd`.

`nettl` is the network tracing and logging subsystem, and in the system default configuration starts 3 daemons, `ntl_reader`, `nktl_daemon` and `netfmt`. These are easily disabled by editing `/etc/rc.config.d/nettl`, however you will lose potentially valuable log data, such as link down messages:

```
Apr  1 12:47:04 bastion vmunix: btlan: NOTE: MII Link Status Not OK - Check Cable Connecti
```

Also, by default console logging is enabled. I find little value in log messages being written to a console that is rarely looked at or may in fact be non-existent. We can disable console logging which causes the console filter formatter daemon, `netfmt` to not start:

```
# nettlconf -L -console 0
# nettl -stop
# nettl -start
Initializing Network Tracing and Logging...
Done.
```

The `nettlconf` command modifies the `nettl` configuration file, `/etc/nettlgen.conf`, so this change will persist across system starts.

`pwgrd` is a password and group caching daemon. Since we have a very small password and group file it is unnecessary. Also, a little detective work with `lsof` and `tusc` (Trace Unix System Calls) [10] shows us that it listens on a Unix domain socket for client requests, and we don't want to allow command channels like that to processes running as root, so we have additional incentive to disable it:

Set the PWGR environment variable to 0 in `/etc/rc.config.d/pwgr`:

```
PWGR=0
```

We also remove stale sockets which will prevent unnecessary libc socket creation and requests to a nonexistent `pwgrd` listener:

```
# rm /var/spool/pwgr/*  # really just need to remove status
# rm /var/spool/sockets/pwgr/*
```

`ptydaemon` is a mystery, since it does not have a man page. A little more detective work leads us to the belief that it may only be used by `vtydaemon`, which we are not using. We decide to kill it and see if we can still login to the system remotely (we temporarily enable `telnetd` to test this). This works fine, so we decide to permanently disable the startup of `ptydaemon`:

Set the PTYDAEMON_START environment variable to 0 in `/etc/rc.config.d/ptydaemon`:

```
PTYDAEMON_START=0
```

Cleanup old logfile:

```
# rm /var/adm/ptydaemonlog
```

8. Examine Set-id Programs

Many Unix systems, including HP-UX, ship with numerous programs that are set-uid or set-gid. Many of these programs are not used, or are only used by the root user. Many of the vulnerabilities that are discovered in Unix utilities rely on the set-uid root bit to raise privilege. You can improve the security of your system by removing these programs or by removing the set-id bit. To obtain a list of all files with either the set-uid or set-gid bit set on the system you can execute:

```
# find / \( -perm -4000 -o -perm -2000 \) -type f -exec ls -ld {} \;
```

You'll probably see well over 100 or so files listed (in the sample configuration there are 145). You may notice that there are two sets of LVM commands (in `/sbin/` and `/usr/sbin/`), each with greater than 25 links, which are set-uid root. Also, the SD commands are set-uid root. The following permission changes will greatly reduce the size of your set-id list:

```
# chmod u-s /usr/sbin/swinstall
# chmod u-s /usr/sbin/vgcreate
# chmod u-s /sbin/vgcreate
```

You will also notice that there are some shared libs that have the set-uid bit set; the reason for this is unknown, however it is safe to remove them. If you did not previously remove all saved patch files in `/var/adm/sw/save/`, you may be surprised to see that they have retained their set-id privilege. While this practice is questionable, they are protected from being executable by non-root users due to the 500 mode on the `/var/adm/sw/save/` directory.

Our strategy is to remove the set-id bits from all files, then selectively add it back to just a few programs that need to be run by non-root users.

The following commands will remove the set-uid and set-gid bits from *all* files, then add it back to `su` and the archive linked version of the `passwd` command:

```
# find / -perm -4000 -type f -exec chmod u-s {} \;
# find / -perm -2000 -type f -exec chmod g-s {} \;
```

```
# chmod u+s /usr/bin/su
# chmod u+s /sbin/passwd
```

The commands you choose to leave set-id depend on the specific usage and policies of your bastion host. Let's say that the bastion host is a firewall gateway, where a few administrators will login via a unique, personal login, then `su` to root to manage the gateway. Here, `/usr/bin/su` may be the only program on the system that needs to be set-uid.

Additionally, a number of commands will function fine without privilege using default or commonly used options, including `bdf`, `uptime` and `arp`--however some functionality may be lost for non-root users. For example, you can no longer specify a filesystem argument for `bdf`:

```
$ bdf /dev/vg00/lvol3
bdf: /dev/vg00/lvol3: Permission denied
```

9. Examine File Permissions

A freshly installed HP-UX system will contain a number of files which are writable by other (the 002 bit is set in the mode bits). These files can be listed with the following:

```
# find / -perm -002 ! -type l -exec ls -ld {} \;
```

We don't display symbolic links with the write other bit set because the mode bits are not used for permission checking.

One approach is to remove the write other bit from all files then selectively add it back to those files and directories where it is necessary. The following can be executed to remove the write other bit from all files with it set:

```
# find / -perm -002 ! -type l -exec chmod o-w {} \;
```

Now we open up the permissions of files that need to be writable by other users:

```
# chmod 1777 /tmp /var/tmp /var/preserve
# chmod 666 /dev/null
```

Note that we also set the sticky bit (01000) in publicly writable directories like `/tmp` and `/usr/tmp`. This prevents unprivileged users from removing or renaming files in the directory that are not owned by them (see `chmod(2)`).

10. Security Network Tuning

HP-UX 11 introduces the `ndd` command to perform network tuning. `ndd -h` produces a list of help text for each supported and unsupported `ndd` tunable parameter that can be changed. After examining this list, we decide the following are candidates for changing on a bastion host:

Network device	Parameter	Default value	Suggested value	Comment
/dev/ip	ip_forward_directed_broadcasts	1	0	Don't forward directed broadcasts
/dev/ip	ip_forward_src_routed	1	0	Don't forward packets with source route options
/dev/ip	ip_forwarding	2	0	Disable IP forwarding
/dev/ip	ip_ire_gw_probe	1	0	Disable dead gateway detection (currently no ndd help text; echo-requests interact badly with firewalls)
/dev/ip	ip_pmtu_strategy	2	1	Don't use echo-request PMTU strategy (can be used for amplification attacks and we don't want to send echo-requests anyway)
/dev/ip	ip_send_redirects	1	0	Don't send ICMP redirect messages (if we have no need to send redirects)

/dev/ip	ip_send_source_quench	1	0	Don't send ICMP source quench messages (deprecated)
/dev/tcp	tcp_conn_request_max	20	500	Increase TCP listen queue maximum (performance)
/dev/tcp	tcp_syn_rcvd_max	500	500	HP SYN flood defense.
/dev/ip	ip_check_subnet_addr	1	0	Permit 0 in local network part (should be the default)
/dev/ip	ip_respond_to_address_mask_broadcast	0	0	Don't respond to ICMP address mask request broadcasts
/dev/ip	ip_respond_to_echo_broadcast	1	0	Don't respond to ICMP echo request broadcasts
/dev/ip	ip_respond_to_timestamp_broadcast	0	0	Don't respond to ICMP timestamp request broadcasts
/dev/ip	ip_respond_to_timestamp	0	0	Don't respond to ICMP timestamp requests

Some of the default values match our preferred value, but we can choose to set them anyway, just in case the default should change in a future release. `ndd` supports a `-c` option which reads a list of tunables and values from the file `/etc/rc.config.d/nddconf`, and which is run automatically at boot time. However, there are some problems with the default setup. First, at the time of this writing, `ndd -c` is only able to handle 10 tunables in `nddconf`. Next, `ndd -c` is run at the end of the `net` script, which is after network interfaces have been configured. One issue with this is it is too late to set `ip_check_subnet_addr` if we are using subnet zero in the local part of a network. But more importantly, we want to set tunables *before* the network interfaces are configured (note: the ordering problem has been fixed in a recent transport patch, but the 10 tunable limit remains).

A workaround is presented that uses a new startup script and configuration file:

```
# cp /tmp/secconf /etc/rc.config.d
# chmod 444 /etc/rc.config.d/secconf
# cp /tmp/sectune /sbin/init.d
# chmod 555 /sbin/init.d/sectune
# ln -s /sbin/init.d/sectune /sbin/rc2.d/S009sectune
```

We run the script immediately after `net.init`, which sets up the plumbing for the IP stack, then runs `ndd -a` which sets transport stack tunable parameters to their default value.

`sectune` and a sample `secconf` are available for download [11].

11. Install Software and Test Configuration

At this point you can install, test and configure the application software that you will use on the bastion host, such as the BIND product, a web server, a firewall product etc. Security software, such as SSH (Secure Shell) and TCP wrappers can be installed at this point, as determined by the specific security requirements and use of the bastion host. Again, extreme caution should be exercised when installing new software on your bastion host. You should generally get the latest version of the product, that has been patched against all known security defects. You may want to install the product first on another system and determine if it can be secured. Think like an attacker, and ensure that the bastion host is able to protect itself with the product installed.

12. Create System Recovery Tape

Next we create a bootable System Recovery Tape of the root volume group; this tape can also be used to clone the system to other hardware that is supported with the same software configuration (for example I can clone from an L2000 to an N4000).

The following can be executed online (very cool), though I gather you will want the system in a somewhat quiescent state:

```
# /opt/ignite/bin/make_recovery -Ai
      Option -A specified. Entire Core Volume Group/disk will be backed up.


      *******************************************
       HP-UX System Recovery
```

```
Going to create the tape.
System Recovery Tape successfully created.
```

Conclusion

With the simple methodology presented, a paranoid mindset, a little detective work and some persistence, it's relatively straightforward to construct a robust bastion host using HP-UX.

References

[1] Marcus J. Ranum, "Thinking About Firewalls", SANS 1993. An updated version, "Thinking About Firewalls V2.0: Beyond Perimeter Security", is available at http://www.clark.net/pub/mjr/pubs/think/index.htm

[2] D. Brent Chapman and Elizabeth D. Zwicky, "Building Internet Firewalls", O'Reilly & Associates, September 1995.

[3] HP-UX patches are available via anonymous FTP in North America at ftp://us-ffs.external.hp.com/hp-ux_patches/; and Europe at ftp://europe-ffs.external.hp.com/hp-ux_patches/.

[4] HP-UX Patch Security Matrix, ftp://europe-ffs.external.hp.com/export/patches/hp-ux_patch_matrix.

[5] HP-UX Patch Checksum Information, ftp://europe-ffs.external.hp.com/export/patches/hp-ux_patch_sums.

[6] HP Security Bulletins are available at http://us-support.external.hp.com/ and http://europe-support.external.hp.com/. Select "Search Technical Knowledge Base" (unfortunately you need a login to access security bulletins, but you can register for one in a few minutes).

[7] Titan host security tool, http://www.fish.com/titan/.

[8] Vic Abell's lsof (LiSt Open Files), ftp://vic.cc.purdue.edu/pub/tools/unix/lsof/.

[9] swagentdk script, http://people.hp.se/stevesk/swagentdk.

[10] tusc (Trace Unix System Calls), syscall tracer for HP-UX, ftp://ftp.cup.hp.com/dist/networking/misc/tusc.shar.

[11] Sample secconf and sectune scripts, http://people.hp.se/stevesk/secconf and http://people.hp.se/stevesk/sectune.

HP ORACLE DATABASE MACHINE

FEATURES AND FACTS

FULL RACK FEATURES

- 8 HP Database Servers: (DL360 G5 servers)
- 14 Exadata Storage Servers (software sold separately)
- 4 Infiniband Switches
- Oracle Database 11g Enterprise Edition with Real Application Clusters and Partitioning (sold separately)
- Oracle Enterprise Linux
- Integrated in one standard rack

FACTS

- User data capacity of up to 21TB (SAS) or up to 46TB (SATA) per rack [1]
- Raw storage capacity of 75TB (SAS) or 168TB (SATA) per rack
- I/O bandwidth of up to 14GB/sec (SAS) or up to 10.5GB/sec (SATA) per rack
- Scale by adding HP Oracle Database Machines; up to 8 machines can be clustered without requiring additional Infiniband switches
- Pre-configured system optimized for data warehousing
- Four InfiniBand network ports are available for connecting to external servers for very fast connectivity to data sources or application servers

RELATED PRODUCTS:

- Oracle Advanced Compression
- Oracle OLAP
- Oracle Data Mining
- Oracle Warehouse Builder
- Oracle BI Suite

RELATED SERVICES

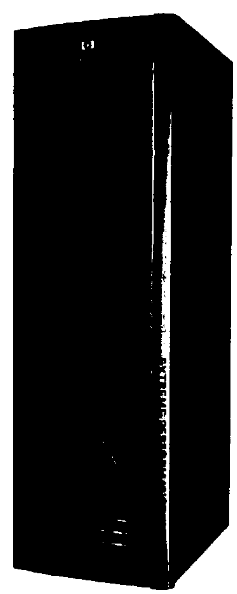
- Oracle Premier Support
- Oracle Consulting

The HP Oracle Database Machine is built for extreme performance for data warehouses. Designed for multi-terabyte data warehouses with I/O-intensive workloads and built using industry-standard hardware from HP and smart storage software from Oracle, the HP Oracle Database Machine is a complete, optimized and preconfigured package of software, servers, and storage. Simple and fast to implement, the HP Oracle Database Machine is ready to tackle your large-scale business queries immediately—and often run them 10x faster or more.

Extreme Performance for Data Warehousing

The HP Oracle Database Machine delivers extreme performance gains for large-scale data warehouse queries, but what does this mean for your business? Questions that used to take hours now only require minutes or seconds. An analyst who could previously ask 2-3 questions per day can now ask dozens of questions per day. Terabytes to petabytes of data that were previously impossible to manage and analyze can now become important information to feed into your business processes.

The unique technology driving these performance gains is the Exadata Storage Server. As data volumes have continued to grow exponentially, conventional storage arrays have struggled to efficiently read terabytes of data from disks, and push that data through storage networks to achieve the performance necessary for IO-intensive data warehouse operations. The HP Oracle Database Machine includes 14 Exadata Storage Servers that provide a high-bandwidth, massively parallel storage solution, delivering up to 14 GB per second of raw I/O bandwidth. When accessing compressed data, the effective throughput for a single rack can be 50 GB per second or more. Each Exadata Storage Server has "Smart Scan" software built-in to execute rapid table scans, sending only required rows and columns through a fast Infiniband network for processing by database servers. This eliminates network bottlenecks and frees database server resources so you can run even more queries faster.



HP Oracle Database Machine Full Rack

ORACLE

FEATURES AND FACTS

HALF RACK FEATURES

- 4 HP Database Servers: (DL360 G5 servers)
- 7 Exadata Storage Servers (software sold separately)
- 2 Infiniband Switches
- Oracle Database 11g Enterprise Edition with Real Application Clusters and Partitioning (sold separately)
- Oracle Enterprise Linux
- Integrated in one standard rack

FACTS

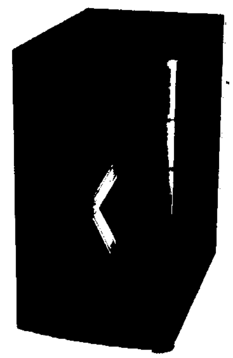
- User data capacity of up to 10.5TB (SAS) or up to 23TB (SATA) per rack [1]
- Raw storage capacity of 38TB (SAS) or 84TB (SATA) per rack
- I/O bandwidth of up to 7GB/sec (SAS) or up to 5.5GB/sec (SATA) per rack
- Two Half Racks can be clustered for larger systems
- Preconfigured system optimized for data warehousing
- Four InfiniBand network ports are available for connecting to external servers for very fast connectivity to data sources or application servers

KEY BENEFITS:

- Extreme performance
- Enterprise ready

Scaling to Petabytes

Utilizing a building-block methodology, the HP Oracle Database Machine provides a quick and easy way to scale. As new racks of HP Oracle Database Machines are incrementally added to a system, the storage capacity and performance of the system grows; a two-rack system is simply twice as powerful as a single rack. Scaling out is easy; the additional HP Oracle Database Machine is connected to the InfiniBand interconnect in existing racks, and Oracle automatically rebalances the database to fully utilize all of the storage and processing power of all racks. Up to eight Database Machines can be networked together without requiring additional InfiniBand switches, and larger configurations can be built with the addition of external switches.


HP Oracle Database Machine Half Rack

Smaller configurations can be supported with the HP Oracle Database Machine Half Rack. With exactly half as many servers as the full-rack Database Machine, the Half Rack is based on a 22U cabinet with 4 database servers, 7 Exadata Storage Servers and two InfiniBand switches. Two Half Racks can be combined to scale for larger data volumes.

Enterprise Ready

Oracle Database is proven to be the database of choice for customers' data warehousing applications, and the HP Oracle Database Machine is built upon Oracle Database 11*g* and Oracle Real Application Clusters. While large I/O bandwidth is crucially important for a successful data warehouse, other ingredients are necessary as well. Oracle Database 11*g* provides the complete platform for data warehousing and business intelligence, with highly sophisticated features, such as:

- Bitmap indexes, advanced partitioning, and multidimensional cubes
- Advanced analytical capabilities with OLAP, data mining, and statistics
- Data integration and data quality capabilities
- Workload management

The HP Oracle Database Machine can support the demands of your business users 24/7, with complete redundancy built in throughout. This ensures that the system remains continuously available in the case of disk, database server, storage server, or network switch failures.

Hardware from HP, Software from Oracle,

The HP Oracle Database Machine is a joint engineering development effort between HP and Oracle that builds upon years of solving customers' business and technical challenges. By combining leading, industry-standard hardware from HP with the intelligence built into Oracle Database 11*g* and the Oracle Exadata Storage Server Software, the HP Oracle Database Machine delivers the highest levels of performance and support across both hardware and software.

HP Oracle Database Machine Hardware Full Rack (all hardware components in a 42U rack)	HP Oracle Database Machine Hardware Half Rack (all hardware components in a 22U rack)
8–HP Proliant DL360 G5 database servers, with	4–HP Proliant DL360 G5 database servers, with
• 2 quad-core Intel Xeon Processor E5430 (2.66GHz)	• 2 quad-core Intel Xeon Processor E5430 (2.66GHz)
• 32GB memory	• 32GB memory
• 1–HP InfiniBand Dual Port HCA	• 1–HP InfiniBand Dual Port HCA
• 4–146GB SAS 10K hard disk drives	• 4–146GB SAS 10K hard disk drives
4–24-port InfiniBand switches	2–24-port InfiniBand switches
14–HP Exadata Storage Server Hardware—each is an HP ProLiant DL180 G5, with	7–HP Exadata Storage Server Hardware—each is an HP ProLiant DL180 G5, with
• 2 quad-core Intel Xeon Processor E5430 (2.66GHz)	• 2 quad-core Intel Xeon Processor E5430 (2.66GHz)
• 8GB memory	• 8GB memory
• 1–HP InfiniBand Dual Port HCA	• 1–HP InfiniBand Dual Port HCA
• 12–450GB SAS or 12–1TB SATA disk drives	• 12–450GB SAS or 12–1TB SATA disk drives
1 48-port Gigabit Ethernet switch (used for management connectivity to servers and switches)	1 48-port Gigabit Ethernet switch (used for management connectivity to servers and switches)
Keyboard, video, mouse (KVM) hardware	Keyboard, video, mouse (KVM) hardware
Oracle Enterprise Linux Release 5.1	Oracle Enterprise Linux Release 5.1
HP Hardware Warranty, 3 year parts/3 year labor/3 year onsite, 4 hour, 24x7	HP Hardware Warranty, 3 year parts/3 year labor/3 year onsite, 4 hour, 24x7
On-site installation and configuration	On-site installation and configuration
Specifications	
• Height: 42U	• Height: 22U
• Width: 613 mm (with side panels)	• Width: 613 mm (with side panels)
• Depth: Front to rear door: 1,015 mm	• Depth: Front to rear door: 1,015 mm
• Flooring requirements: 1350 lbs	• Flooring requirements: 820 lbs
• System inlet temperature (Operating): 10° to 35° C (50° to 95° F) at sea level	• System inlet temperature (Operating): 10° to 35° C (50° to 95° F) at sea level
Key Capabilities (per rack)	
HP Oracle Database Machine Hardware Full Rack SAS 450	HP Oracle Database Machine Hardware Half Rack SAS 450
• Up to 14 GB/sec of raw, uncompressed I/O throughput	• Up to 7 GB/sec of raw, uncompressed I/O throughput
• Up to 1 TB/hour data loading	• Up to 500 GB/hour data loading
• Up to 21 TB of user data[1]	• Up to 10.5 TB of user data[1]
HP Oracle Database Machine Hardware Full Rack SATA 1000	HP Oracle Database Machine Hardware Half Rack SATA 1000
• Up to 10.5 GB/sec of raw, uncompressed I/O throughput	• Up to 5.3 GB/sec of raw, uncompressed I/O throughput
• Up to 1 TB/hour data loading	• Up to 500 GB/hour data loading
• Up to 46 TB of user data[1]	• Up to 23 TB of user data[1]

[1] User data capacity is computed after mirroring and after allowing space for database structures such as temp, logs, undo, and indexes. User data capacity is uncompressed; with compression, 2x to 4x more data can often be stored. Actual user data capacity varies by application.

HP Oracle Database Software (sold separately)
• For database servers: Oracle Database 11g Enterprise Edition, Oracle Real Application Clusters and Oracle Partitioning
• For storage servers: Oracle Exadata Storage Server Software

High-Availability Features
• Redundant power supplies for all servers
• Redundant InfiniBand switches
• Oracle Automatic Storage Management: All database files mirrored; disk failures do not interrupt query processing
• Oracle Real Application Clusters: database server failures are tolerated
• Oracle Exadata Storage Server Software: storage server failures are tolerated
• Backup is performed using Oracle Recovery Manager
• Point in time restores are performed using Oracle Flashback Technologies
Manageability Features
• HP Lights-Out hardware management
• Oracle Enterprise Manager Database Control

Contact Us

For more information about the HP Oracle Database Machine please visit oracle.com/exadata or call +1.800.ORACLE1 to speak to an Oracle representative.

THE WALL STREET JOURNAL.
WSJ.com

INVESTING | JULY 17, 2009

The Internet Is Dead (As An Investment)

By James Altucher | A Dow Jones Newswires Column

I can live all day inside the Internet. I can talk to my friends, listen to music, watch TV, trade stocks, play games, do work - all on the Internet. From 6 a.m. until 10 p.m. every day I can spend on the Internet and it would be a day well spent.

But run for the hills when it comes to advising clients to invest in the Internet.

The days of infinite margins, 1,000% productivity gains, and growth of market throughout the universe are long over. Internet companies now should be treated, at best, like utility companies that get bought at about 10 times earnings and sold at 13 times earnings. Even then, I'm not sure I would give the Internet sector the same respect as the monopoly-protected utility sector.

Don't just ask me. Ask the best. Nobody can figure out a business model.

Time Warner would rather keep their legacy old-media businesses like People magazine than hold onto one of the biggest Internet companies out there, AOL. And News Corp. is shaking up its MySpace business as it figures out its next steps. (News Corp. owns Dow Jones, publisher of this newswire.) Microsoft has spent billions on Internet strategy without a dime of profit. And even Google can't seem to find any other business model other than the one they stumbled into when they bought Applied Semantics in 2001 that had a little piece of software called AdSense. And the new guys: Twitter and Facebook are still scrambling for profits despite blistering usage growth.

What about the nuts-and-bolts guys? Cisco, at 15 times earnings, trades in line with the S&P 500. Buy them when they start giving a steady dividend.

Let's face it. Electricity greatly improved our quality of life. But I'm not going to get excited about buying a basket of utility companies. Same for the Internet. Can't live without it, but can't live with it (in my portfolio).

So what do we do?

In this economy, it's back to the basics. Regardless of how you feel about $1 trillion in stimulus (with more probably on the way), the best growth is going to come from the companies that help us spend that stimulus.

Check out LNN, Lindsay Corporation, that does boring stuff like highway repair (they make those orange cones) and helps upgrade water infrastructure. With half of all hospital beds in the world filled by people with dirty water-related illnesses, this one is a good bet.

Or little known Colfax Corporation, CFX. At nine times forward earnings, this company is in the "fluid handling" business. Boring. But in a resource-starved world we need them to get oil quickly through the pipelines and into the refineries. And we can't forget about ASTE, Astec Industries, which is like the "Amazon of Asphalt" and is a major player in highway repair (think stimulus again).

The exciting plays right now are the companies that are rebuilding the country along with the economy. Save the Internet for your iTunes downloads. But focus client portfolios on the future. Next article: my favorite biotech plays.

James Altucher is a managing partner of Formula Capital, an alternative asset management firm, and an author on investment strategies. Unlike Dow Jones reporters, he may have positions in the stocks he writes about.

Google to Challenge Microsoft With Operating System

By Brian Womack

July 8 (Bloomberg) -- **Google Inc.**, owner of the most- visited Internet search engine, is developing a computer operating system based on its Chrome Web browser, taking aim at **Microsoft Corp.** in its strongest market.

The system will be designed at first for low-cost laptops called netbooks, Google said in a **blog post**. The company is in talks with partners on the project and computers running the software will be available in the second half of 2010.

The plan escalates Google's rivalry with Microsoft, which extends to Web search, browsers and business applications such as word processing and spreadsheets. Windows, Microsoft's flagship product, runs about 90 percent of the world's personal computers. Google is also trying to spur Web-ad sales after reporting its first sequential revenue drop as a public company.

"There is a possibility that the new as can break the paradigm Microsoft and Intel created over the past 20 years," aid **Yukihiko Shimada**, a computer analyst at Mitsubishi UFJ Securities Co. in Tokyo. "There is plenty of business opportunity for Google in this market."

Google said it's working with computer makers to introduce a number of netbooks next year, without identifying any of the companies. The Chrome OS will be open-source, meaning the program code will be open to developers, Google said. The software will work on top of the Linux Operating system.

Netbook Competition

Frank Shaw, a spokesman for Redmond, Washington-based Microsoft, declined to comment. Windows accounted for 28 percent of the company's $60.4 billion **annual revenue** in the 12 months ended June 30, 2008.

Microsoft has stepped up its efforts in the netbook market. It said in May it plans to remove a restriction of running three applications at a time on its forthcoming Windows 7 Starter Edition, which is designed for netbooks. The announcement eliminated one of the most significant differences between the basic edition of the operating system and a pricier one.

Google, based in Mountain View, California, rose $5.86, or 1.5 percent, to 402.49 in Nasdaq Stock Market trading at 4 p.m. New York time. Microsoft advanced 3 cents $22.56. Google has risen 31 percent this year and **Microsoft** has added 16 percent.

Online Strategy

X1 The Chrome OS is consistent with Google's focus on getting people

to use software online, which contrasts with Microsoft's approach of providing programs on the computer itself. Google started offering business software in 2007, allowing users to access spreadsheets and word-processing documents via the Web, just as anyone might access the search engine or Google News.

"We hear a lot from our users and their message is clear -- computers need to get better," Google said. "The operating systems that browsers run on were designed in an era where there was no Web."

Google is trying to encourage people to spend more time online to fuel demand for Internet ads, which accounted for more than 90 percent of its 2008 revenue of $21.8 billion. In the first quarter of this year, Google had its first sequential sales drop since it went public in 2004 as the recession prompted companies to curb advertising spending.

The open-source Chrome as will probably win over companies that don't want to pay for Windows, said Jim Friedland, an analyst at Cowen & Co. in New York. The system may also help Google sell Web-based applications, such as the Gmail e-mail service, that run on top of it, he said.

"They really haven't cracked the enterprise yet," said Friedland, who rates Google's shares "outperform" and doesn't own any. "We've seen some inroads around the edges."

More Options

Computer makers such as Acer Inc. and Asustek Computer Inc. already have plans to offer computers running Android, an open - source operating System backed by Google and initially designed for mobile phones. Acer, the world's second-largest laptop maker, said last month it plans to release a low-cost notebook powered by Android. Asustek Computer has also developed a netbook that runs on Google's software.

"Having another as or another interface does create more options, and with the weight of the Google name behind it, does lift its prominence," said Bryan Ma, a computer analyst at IDC in Singapore.

Google said that while the Chrome as is separate from Android, the two will overlap in some areas. The Chrome OS operating system is designed to save users from having to deal with viruses and security updates, Google said.

"Google Chrome as is being created for people who spend most of their time on the Web, and is being designed to power computers ranging from small netbooks to full-size desktop systems," Google said. "While there are areas where Google Chrome as and Android overlap, we believe choice will drive innovation for the benefit of Google."

Beneficial to Users

Hewlett-Packard is studying the Chrome OS and is open to "various approaches to meet its own customer needs," Marlene Somsak, a spokeswoman for the company, said in an e-mail. She declined to comment on whether Hewlett-Packard plans to introduce products based on the system.

Dell constantly assesses new technology as part of its product-development process, said Anne Camden, a spokeswoman for the company.

Tony Chen, chief operating officer of Asustek's notebook unit said by phone the company will consider "anything, that's beneficial to users." Fujitsu Ltd. spokeswoman Nozomi Endo said the company will monitor market conditions before deciding whether to introduce products using Google's operating system.

Spokespeople for Acer, Sony Corp., Samsung Electronics Co., NEC Corp., Panasonic Corp., and Toshiba Corp, declined to comment.

Search Rivalry

The Chrome OS -- which will run on traditional Intel Corp.- based x86 chips along with semiconductors design by ARM Holdings Plc -- will work on lightweight netbooks along with more powerful computers, including desktop PCs, Google said.

Google's Chrome still faces an uphill battle against Microsoft's browser. Chrome, which was unveiled last year, had 1.2 percent market share in February, compared with 67 percent for Microsoft's Internet Explorer, according to research firm Net Applications, which tracks Web statistics.

In May, Microsoft introduced a search engine called Bing that has enhanced shopping, travel and sorting features. Bing's market share climbed to more than 10 percent in June, according to Comscore Inc.

Google's search engine is No.1 in the U.S., holding more than 60 percent market share. Microsoft is No.3, according to ComScore.

THE WALL STREET JOURNAL.
WSJ.com

SUBSCRIBER CONTENT PREVIEW

TECHNOLOGY | AUGUST 5, 2009, 7:22 P.M. ET

Google to Acquire Video-Technology Firm

A WALL STREET JOURNAL ONLINE NEWS ROUNDUP

Google Inc. said it will issue $106.5 million in stock to acquire On2 Technologies Inc. as the Internet-search giant looks to buttress its video operations.

On2 makes video-compression technology, which allows for the quick transfer of large video files across the Web. On2 also has software that helps make high-definition video playback possible on mobile devices.

Google, based in Mountain View, Calif., has one of the most-visited video sites, YouTube.

"We are committed to innovation in video quality on the web, ...

THE WALL STREET JOURNAL.
WSJ.com

OPINION: THE WEEKEND INTERVIEW | JULY 24, 2009, 10:58 P.M. ET

Silicon Valley Survivor

Cisco's CEO says Internet 2.0 will drive productivity growth in the new, new economy.

John Chambers By MICHAEL S. MALONE

San Jose, Calif.

The crash has hit Silicon Valley as hard as anywhere else. The only consolation is that this time, at least, it didn't start here. But while other firms are hunkering down and trying to survive, networking giant Cisco Systems ($130 billion market value, $40 billion in annual sales) continues to stride forward.

"Even in this downturn," says chairman and CEO John Chambers, sitting at the table in his modest, memento-crammed office, "We intend to be the most aggressive we've ever been."

A decade ago Cisco was known for building the switching systems (called routers) that find the most efficient path for information on the Internet. These days it has a presence in just about every corner of the Web's infrastructure—from networking hardware (switches, gateways and the like) to network management software (CiscoWorks) to the Linksys router that runs the wireless network in your home.

As much as three-quarters of the world's digital data now passes through Cisco equipment, the company estimates. This massive expansion has come not merely by internal growth, but through one of the biggest (and most successful) merger-and-acquisition programs in high-tech history.

Despite taking the same hit as the rest of the computer industry earlier this year, Cisco's stock is climbing (up 17% to more than $21.50 in the last month), the company has cut an incredible $1.5 billion in operating expenses, and earned an upgrade this week to "outperform" from Credit Suisse. It has also announced a major sponsorship of the 2012 Olympic Games in London.

Cisco is feeling so confident that it is even planning a new integrated hardware/software platform (Cisco Unified Computing System) that puts it in direct competition with its former strategic partners, IBM and Hewlett-Packard, for control of the multibillion dollar business of equipping the thousands of data centers run by large corporations.

Meanwhile, Cisco has had no major layoffs during this economic downturn and made no salary cuts. Six weeks ago the company announced its intention to pursue 30 disparate new market areas—from sports to the smart grid (delivering electricity efficiently through digital technology) to cloud computing (using thousands of computers to share work)—simultaneously.

"You never want to waste a good crisis," Mr. Chambers says with a smile, and he has publicly predicted an average 12% to 17% annual growth over the course of the next five years whatever happens to the economy in the short term.

"We basically wrote the press release for five years from now," he says, "and now we plan to live up to it." Mighty bold talk. Chatting with Mr. Chambers, I find it easy to forget we're in the worst recession in a generation. To hear him tell it, however, this is the perfect moment for Cisco to outrun the competition, just as it did during other crises in the often feast-or-famine world of high tech. "Each time we emerged out of it a stronger company, with more market share, and we moved into more [adjoining] markets," he says.

Yet Mr. Chambers wasn't always so optimistic. Those who watched him on television in early 2001 when the dot-com bubble burst—and who could read the signals (his courtly Southern accent going flat and unemotional, his natural ebullience replaced with a dry delivery of facts)—saw a man looking into his own career grave for the second time.

The son of an obstetrician/gynecologist father and a psychologist mother who also owned a restaurant, John Chambers was born in Cleveland but raised in West Virginia. That explains both his accent (rare in Silicon Valley) and his love of the outdoors. "I grew up fishing, hunting and swimming," he says. "I lived in the river." In fact, after this conversation he was set to take, on his own dime, 20 company executives on a fishing trip to Alaska.

Mr. Chambers credits his parents with giving him a doctor's approach to business problems. "They taught me to focus on the long-term health of the patient, and not to treat symptoms but the underlying problem." They also gave him an obsession with maintaining his health, giving him advice he admits to ignoring as a young man. "My father told me not to ride motorcycles or jump of bridges—both of which I foolishly ignored." He's more careful these days, although he recently learned how to fly a helicopter.

As is mentioned in every biography of the man, John Chambers is a severe dyslexic, although that didn't keep him from earning a law degree from West Virginia University and an MBA from Indiana University. "He had this very optimistic attitude about everything," his childhood reading therapist would later say. "He was just not going to fail. One thing I notice as I hear him now on TV is that he still has that attitude."

Mr Chambers never practiced law. Instead, seeing the future in computers and business, he joined IBM in 1976 as a salesman. Seven years later, he went to work at minicomputer maker Wang Laboratories, where his career climb was meteoric: By 1990 he was executive vice president.

It was at Wang—a billion-dollar company that seemed to die almost overnight in 1990 because of bad management and a failure to adapt to the personal computer revolution—where Mr. Chambers faced his first career crisis. "I felt I knew what we needed to do, but I wasn't able to convince Dr. Wang to take the risk," he says mournfully. He says he is still haunted by the cost: "37,000 people lost their jobs, and the shareholders lost everything."

In 2001, five years into his tenure as CEO of Cisco (the only company that answered his job application after Wang), Mr. Chambers stared disaster in the face again. In the fall, Cisco, with a market capitalization of $500 billion, was the world's most valuable corporation. As he departed for a Christmas vacation, the company had a 26-week backlog of orders and was enjoying 70% annual growth.

He returned to an industry that had collapsed. Orders had disappeared, inventories had evaporated, and by the middle of January, he recalls with a shudder, "the 70% growth went to minus 45%. That's a drop of 115%. I thought that was mathematically impossible."

But, under Mr. Chambers's leadership, Cisco kept its cool. There were losses and there were layoffs, but after the cuts Cisco gained market share against its even more damaged competitors—and hasn't stopped.

Cisco's growth plan has combined audacity in acquisitions and attacking new markets with strict, even ruthless, control over

costs. "We've learned our lessons with each downtown in this company's history," Mr. Chambers says, "and we've fine-tuned them as we've gone along. Now we have our playbook, with its four key elements, and we're going to run that game no matter what the short-term situation looks like."

The first element of his playbook, he says, is to "be realistic," that is, to gauge how many challenges are created by the economy, and how many are self-inflicted. The second is to "assess your situation," which means to ask how long a downturn will last and how deep it will be. "And," Mr. Chambers says, "it will usually be longer than you think." His third element is to "get ready for the upturn," and the fourth is to "get closer to your customers."

Says Mr. Chambers: "We added that last one in '97," after Cisco noticed too late that its customers were sliding into a recession.

Mr. Chambers has three more unwritten rules, for good times and bad. The first is "watch the stock market": "When the [dot-com bubble] crash came in 2001, the stock market had been going down for a year, but we hadn't really noticed. I watch it a lot more closely now," he says, an especially good idea after Cisco replaced General Motors on the Dow Jones Industrial Average on June 9 of this year.

The second is to "always have more cash, not less." "We didn't have enough cash going into 2001," Mr. Chambers says. "This time we went into it with $34 billion. That wasn't accidental."

Finally, Mr. Chambers's third unwritten rule is to "be aggressive." Coming out of the 2001 crash, Cisco embarked on one of the most spectacular acquisition programs in high tech history—more than 130 companies to date—in the process expanding its product lines from computer networking routers to local area networking switching, voice-over IP (the technology that allows Internet telephony), and home networks.

High-tech companies are notorious for killing the firms they acquire, but several of Cisco's major acquisitions (including Linksys, Scientific Atlanta and WebEx) have turned into billion dollar businesses. Mr. Chambers credits this success to letting acquired companies continue to do what he bought them for and keeping their management.

Since the 2001 downturn, Cisco has embarked on a radical reorganization that Mr. Chambers believes best positions the company to compete. Cisco was an early adopter of new technologies, pioneering online commerce. By 1999 it had booked 90% of its orders on the Internet.

Now Cisco has kicked into high gear as the world enters what Mr. Chambers calls "Internet 2.0."

X This Internet 2.0, Mr. Chambers says, will be all about pervasive video and collaboration tools. It will power the social networking Web world to even greater heights. Much more importantly for our economy, it will "drive productivity by 2% to 3% per year," he says.

Mr. Chambers plans to have Cisco build and run Internet 2.0. That's why he's turning on old allies such as HP (they used to work together pursuing the data center business) and embracing new ones. That's also why he's embracing video technology, from video cameras to videoconferencing (he "visited" 270 offices around the world last year, 200 of them virtually). And it's why he's reorganized Cisco around small, fast-moving product groups—called councils and boards—and sent them off in hot pursuit of markets in full expectation of every one turning into a billion-dollar business.

Unrolling all of that may take a decade or more, and John Chambers intends to be around to see it happen. He certainly has the genes for it: His father, 84, remains his most trusted adviser.

Mr. Chambers also intends to be happy along the way. He bounds into the interview looking at least a decade younger than his 60 years, and in the course of our conversation pulls out and demonstrates Cisco's new, low-cost HD Flip digital movie camera.

"It's CEO idiot-proof," he laughs.

Mr. Chambers was offered one of the cameras as a gift from the manufacturer, Pure Digital Tech, last year. He refused to accept it unless he paid retail. This March he bought the company for $590 million.

Mr. Malone, a columnist for ABCNews.com, is the author of the recently published "The Future Arrived Yesterday" (Crown Business).

Introducing the Google Chrome OS

7/07/2009 09:37:00 PM

It's been an exciting nine months since we launched the Google Chrome browser. Already, over 30 million people use it regularly. We designed Google Chrome for people who live on the web — searching for information, checking email, catching up on the news, shopping or just staying in touch with friends. However, the operating systems that browsers run on were designed in an era where there was no web. So today, we're announcing a new project that's a natural extension of Google Chrome — the Google Chrome Operating System. It's our attempt to re-think what operating systems should be.

Google Chrome OS is an open source, lightweight operating system that will initially be targeted at netbooks. Later this year we will open-source its code, and netbooks running Google Chrome OS will be available for consumers in the second half of 2010. Because we're already talking to partners about the project, and we'll soon be working with the open source community, we wanted to share our vision now so everyone understands what we are trying to achieve.

Speed, simplicity and security are the key aspects of Google Chrome OS. We're designing the OS to be fast and lightweight, to start up and get you onto the web in a few seconds. The user interface is minimal to stay out of your way, and most of the user experience takes place on the web. And as we did for the Google Chrome browser, we are going back to the basics and completely redesigning the underlying security architecture of the OS so that users don't have to deal with viruses, malware and security updates. It should just work.

Google Chrome OS will run on both x86 as well as ARM chips and we are working with multiple OEMs to bring a number of netbooks to market next year. The software architecture is simple — Google Chrome running

within a new windowing system on top of a Linux kernel. For application developers, the web is the platform. All web-based applications will automatically work and new applications can be written using your favorite web technologies. And of course, these apps will run not only on Google Chrome OS, but on any standards-based browser on Windows, Mac and Linux thereby giving developers the largest user base of any platform.

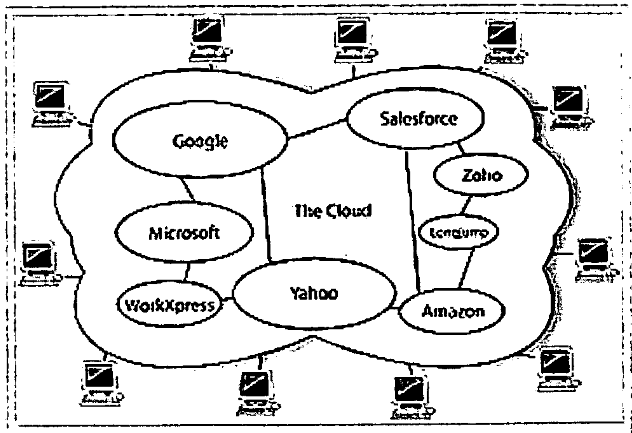
Google Chrome OS is a new project, separate from Android. Android was designed from the beginning to work across a variety of devices from phones to set-top boxes to netbooks. Google Chrome OS is being created for people who spend most of their time on the web, and is being designed to power computers ranging from small netbooks to full-size desktop systems. While there are areas where Google Chrome OS and Android overlap, we believe choice will drive innovation for the benefit of everyone, including Google.

We hear a lot from our users and their message is clear — computers need to get better. People want to get to their email instantly, without wasting time waiting for their computers to boot and browsers to start up. They want their computers to always run as fast as when they first bought them. They want their data to be accessible to them wherever they are and not have to worry about losing their computer or forgetting to back up files. Even more importantly, they don't want to spend hours configuring their computers to work with every new piece of hardware, or have to worry about constant software updates. And any time our users have a better computing experience, Google benefits as well by having happier users who are more likely to spend time on the Internet.

We have a lot of work to do, and we're definitely going to need a lot of help from the open source community to accomplish this vision. We're excited for what's to come and we hope you are too. Stay tuned for more updates in the fall and have a great summer.

Cloud computing

From Wikipedia, the free encyclopedia



Some of the vendors providing Cloud computing services

Cloud computing is a style of computing in which dynamically scalable and often virtualized resources are provided as a service over the Internet.[1][2] Users need not have knowledge of, expertise in, or control over the technology infrastructure in the "cloud" that supports them.[3]

Gartner defines five attributes to Cloud Computing:

- It is service-based.
- It is scalable and elastic. I.e., it is able to add and remove infrastructure as needed.
- It uses shared infrastructure to build economies of scale.
- It is metered and users pay according to usage.
- Most importantly, of course, it uses Internet technologies.

The concept generally incorporates combinations of the following:

- infrastructure as a service (IaaS)
- platform as a service (PaaS)
- software as a service (SaaS)
- Other recent (ca. 2007–09)[4][5] technologies that rely on the Internet to satisfy the computing needs of users. Cloud computing services often provide common business applications online that are accessed from a web browser, while the software and data are stored on the servers.

The term *cloud* is used as a metaphor for the Internet, based on how the Internet is depicted in computer network diagrams and is an abstraction for the complex infrastructure it conceals.[6]

The first academic use of this term appears to be by Prof. Ramnath K. Chellappa (currently at Goizueta Business School, Emory University) who originally defined

it as a *computing paradigm where the boundaries of computing will be determined by economic rationale rather than technical limits.*[7]

Contents

Brief

Comparisons

Cloud computing can be confused with:
1) grid computing - "a form of distributed computing whereby a 'super and virtual computer' is composed of a cluster of networked, loosely coupled computers, acting in concert to perform very large tasks".
2) utility computing - the "packaging of computing resources, such as computation and storage, as a metered service similar to a traditional public utility such as electricity"[8] and
3) autonomic computing - "computer systems capable of self-management".[9]

Indeed, many cloud computing deployments as of 2009 depend on grids, have autonomic characteristics, and bill like utilities—but cloud computing tends to expand what is provided by grids and utilities.[10] Some successful cloud architectures have little or no centralized infrastructure or billing systems whatsoever, including peer-to-peer networks such as BitTorrent and Skype, and volunteer computing such as SETI@home.[11][12]

Furthermore, many analysts are keen to stress the evolutionary, incremental pathway between grid technology and cloud computing, tracing roots back to Application Service Providers (ASPs) in the 1990s and the parallels to SaaS, often referred to as applications on the cloud. [13] Some are of the persuasion that the true difference between these terms is marketing and branding; that the technology evolution was incremental and the marketing evolution discrete. [14]

Characteristics

Cloud computing customers do not generally own the physical infrastructure serving as host to the software platform in question. Instead, they avoid capital expenditure by renting usage from a third-party provider. They consume resources as a service and pay only for resources that they use. Many cloud-computing offerings employ the utility computing model, which is analogous to how traditional utility services (such as electricity) are consumed, while others bill on a subscription basis. Sharing "perishable and intangible" computing power among

multiple tenants can improve utilization rates, as servers are not unnecessarily left idle (which can reduce costs significantly while increasing the speed of application development). A side effect of this approach is that overall computer usage rises dramatically, as customers do not have to engineer for peak load limits.[15] Additionally, "increased high-speed bandwidth" makes it possible to receive the same response times from centralized infrastructure at other sites.

Economics

Cloud computing users can avoid capital expenditure (CapEx) on hardware, software, and services when they pay a provider only for what they use. Consumption is usually billed on a utility (e.g. resources consumed, like electricity) or subscription (e.g. time based, like a newspaper) basis with little or no upfront cost. A few cloud providers are now beginning to offer the service for a flat monthly fee as opposed to on a utility billing basis. Other benefits of this time sharing style approach are low barriers to entry, shared infrastructure and costs, low management overhead, and immediate access to a broad range of applications.

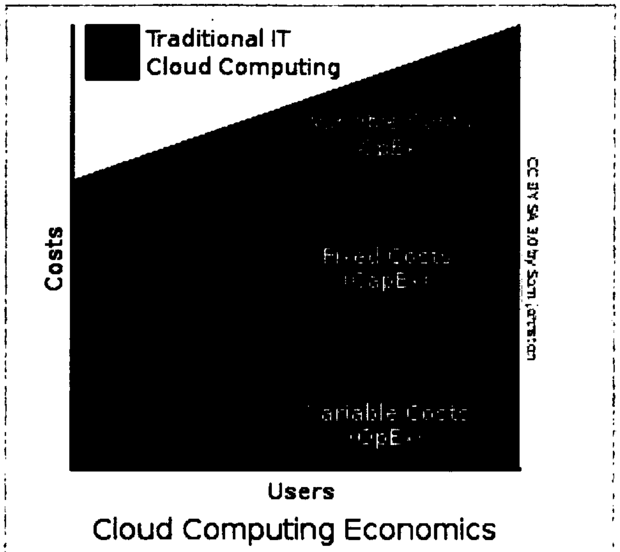


Diagram showing economics of cloud computing versus traditional IT, including capital expenditure (CapEx) and operational expenditure (OpEx)

Users can generally terminate the contract at any time (thereby avoiding return on investment risk and uncertainty) and the services are often covered by service level agreements (SLAs) with financial penalties.[16][17]

According to Nicholas Carr, the strategic importance of information technology is diminishing as it becomes standardized and less expensive. He argues that the cloud computing paradigm shift is similar to the displacement of electricity generators by electricity grids early in the 20th century.[18]

Although companies might be able to save on upfront capital expenditures, they might not save much and might actually pay more for operating expenses. In

situations where the capital expense would be relatively small, or where the organization has more flexibility in their capital budget than their operating budget, the cloud model might not make great fiscal sense. Other factors impacting the scale of any potential cost savings include the efficiency of a company's data center as compared to the cloud vendor's, the company's existing operating costs, the level of adoption of cloud computing, and the type of functionality being hosted in the cloud. [19][20]

Companies

Dell, Vmware, Sun Microsystems, Rackspace US,ThinkGrid, Star UK, IBM, Amazon, Google, BMC, Microsoft and Yahoo are some of the major cloud computing service providers. Cloud services are also being adopted by individual users through large enterprises including Vmware, General Electric, and Procter & Gamble[21][22].

Architecture

The majority of cloud computing infrastructure, as of 2009, consists of reliable services delivered through data centers and built on servers with different levels of virtualization technologies. The services are accessible anywhere that provides access to networking infrastructure. Clouds often appear as single points of access for all consumers' computing needs. Commercial offerings are generally expected to meet quality of service (QoS) requirements of customers and typically offer SLAs.[23] Open standards are critical to the growth of cloud computing, and open source software has provided the foundation for many cloud computing implementations.[24]

History

The Cloud is a term that borrows from telephony. Up to the 1990s, data circuits (including those that carried Internet traffic) were hard-wired between destinations. Subsequently, long-haul telephone companies began offering Virtual Private Network (VPN) service for data communications. Telephone companies were able to offer VPN based services with the same guaranteed bandwidth as fixed circuits at a lower cost because they could switch traffic to balance utilization as they saw fit,

thus utilizing their overall network bandwidth more effectively. As a result of this arrangement, it was impossible to determine in advance precisely which paths the traffic would be routed over. The term "telecom cloud" was used to describe this type of networking, and cloud computing is conceptually somewhat similar.

Cloud computing relies heavily on virtual machines (VMs), which are spawned on demand to meet user needs. A common depiction in network diagrams is a cloud outline.[6]

The underlying concept of cloud computing dates back to 1960, when John McCarthy opined that "computation may someday be organized as a public utility"; indeed it shares characteristics with service bureaus that date back to the 1960s. The term *cloud* had already come into commercial use in the early 1990s to refer to large Asynchronous Transfer Mode (ATM) networks. [25] Ill-fated startup General Magic launched a short-lived cloud computing product in 1995 in partnership with several telecommunications company partners such as AT&T, just before the consumer-oriented Internet became popular. By the turn of the 21st century, the term "cloud computing" began to appear more widely,[26] although most of the focus at that time was limited to SaaS.

In 1999, Salesforce.com was established by Marc Benioff, Parker Harris, and their associates. They applied many technologies developed by companies such as Google and Yahoo! to business applications. They also provided the concept of "On demand" and SaaS with their real business and successful customers. The key for SaaS is that it is customizable by customers with limited technical support required. Business users have enthusiastically welcomed the resulting flexibility and speed.

In the early 2000s, Microsoft extended the concept of SaaS through the development of web services. IBM detailed these concepts in 2001 in the Autonomic Computing Manifesto (http://www.research.ibm.com/autonomic/index.html), which described advanced automation techniques such as self-monitoring, self-healing, self-configuring, and self-optimizing in the management of complex IT systems with heterogeneous storage, servers, applications, networks, security mechanisms, and other system elements that can be virtualized across an enterprise.

Amazon played a key role in the development of cloud computing by modernizing

their data centers after the dot-com bubble and, having found that the new cloud architecture resulted in significant internal efficiency improvements, providing access to their systems through Amazon Web Services in 2005 on a utility computing basis.[27]

In 2007, Google, IBM, and a number of universities embarked on a large scale *cloud computing* research project,[28] around the time the term started, it was a hot topic. By mid-2008, cloud computing gained popularity in the mainstream press, and numerous related events took place.[29]

In August 2008, Gartner Research observed that "organizations are switching from company-owned hardware and software assets to per-use service-based models" and that the "projected shift to cloud computing will result in dramatic growth in IT products in some areas and in significant reductions in other areas."[30]

In 2009, Cloud Computing Solutions by Google, Amazon, Microsoft, and IBM are the most popular among users with Sun and Ubuntu following them in the Cloud.[31]

Criticism and Disadvantages of Cloud Computing

Because cloud computing does not allow users to physically possess the storage of their data (the exception being the possibility that data can be backed up to a user-owned storage device, such as a USB flash drive or hard disk) it does leave responsibility of data storage and control in the hands of the provider.

Cloud computing has been criticized for limiting the freedom of users and making them dependent on the cloud computing provider, and some critics have alleged that it is only possible to use applications or services that the provider is willing to offer. Thus, *The London Times* compares cloud computing to centralized systems of the 1950s and 60s, by which users connected through "dumb" terminals to mainframe computers. Typically, users had no freedom to install new applications and needed approval from administrators to achieve certain tasks. Overall, it limited both freedom and creativity. The Times argues that cloud computing is a regression to that time.[32]

One of the important issues in cloud computing that needs to be addressed is that

once you upload your data to cloud computing service provider, you lose control over your data and if computing service provider is experiencing problems, you may not be able to access your data at all. Also, in most of the cases, at least this is true for free services, there is no one on the provider's side to assist you with a problem (if you are having one).

Similarly, Richard Stallman, founder of the Free Software Foundation, believes that cloud computing endangers liberties because users sacrifice their privacy and personal data to a third party. He stated that cloud computing is "simply a trap aimed at forcing more people to buy into locked, proprietary systems that would cost them more and more over time."[33]

Further to Stallman's observation, It would be a challenge for hosting/deploying intranet and access restricted (for Govt., defense, institutional, etc) sites and their maintenance. Commercial sites using tools such as web analytics may not be able to capture right data for their business planning etc.

Political issues

The Cloud spans many borders and "may be the ultimate form of globalization."[34] As such, it becomes subject to complex geopolitical issues, and providers are pressed to satisfy myriad regulatory environments in order to deliver service to a global market. This dates back to the early days of the Internet, when libertarian thinkers felt that "cyberspace was a distinct place calling for laws and legal institutions of its own"[34].

Despite efforts (such as US-EU Safe Harbor) to harmonize the legal environment, as of 2009, providers such as Amazon Web Services cater to major markets (typically the United States and the European Union) by deploying local infrastructure and allowing customers to select "availability zones."[35] Nonetheless, concerns persist about security and privacy from individual through governmental levels (e.g., the USA PATRIOT Act, the use of national security letters, and the Electronic Communications Privacy Act's *Stored Communications Act*.

Legal issues

In March 2007, Dell applied to trademark the term "cloud computing" (U.S. Trademark 77,139,082 (http://tarr.uspto.gov/servlet/tarr?regser=serial&entry=77139082)) in the United States. The "Notice of Allowance" the company received in July 2008 was canceled in August, resulting in a formal rejection of the trademark application less than a week later.

In September 2008, the United States Patent and Trademark Office (USPTO) issued a "Notice of Allowance" to CGactive LLC (U.S. Trademark 77,355,287 (http://tarr.uspto.gov/servlet/tarr?regser=serial&entry=77355287)) for "CloudOS". As defined under this notice, a *cloud operating system* is a generic operating system that "manage[s] the relationship between software inside the computer and on the Web", such as Microsoft Azure[36].

In November 2007, the Free Software Foundation released the Affero General Public License, a version of GPLv3 intended to close a perceived legal loophole associated with Free software designed to be run over a network, particularly SaaS. An application service provider is required to release any changes they make to Affero GPL open source code.

Risk mitigation

Corporations or end-users wishing to avoid not being able to access their data—or even losing it—are typically advised to research vendors' policies on data security before using their services. One technology analyst and consulting firm, Gartner, lists several security issues that one should discuss with cloud-computing vendors:

- Privileged user access—Who has specialized access to data and about the hiring and management of such administrators?
- Regulatory compliance—Is the vendor willing to undergo external audits and/or security certifications?
- Data location—Does the provider allow for any control over the location of data?
- Data segregation—Is encryption available at all stages, and were these encryption schemes designed and tested by experienced professionals?
- Recovery—What happens to data in the case of a disaster, and does the vendor offer complete restoration, and, if so, how long does that process take?

- Investigative Support—Does the vendor have the ability to investigate any inappropriate or illegal activity?
- Long-term viability—What happens to data if the company goes out of business, and is data returned and in what format?[37]
- Data availability—Can the vendor move your data onto a different environment should the existing environment become compromised or unavailable?

In practice, one can best determine data-recovery capabilities by experiment; for example, by asking to get back old data, seeing how long it takes, and verifying that the checksums match the original data. Determining data security can be more difficult, but one approach is to encrypt the data yourself. If you encrypt data using a trusted algorithm, then, regardless of the service provider's security and encryption policies, the data will only be accessible with the decryption keys. This leads, however, to the problem of managing private keys in a pay-on-demand computing infrastructure.

Key characteristics

- **Agility** improves with users able to rapidly and inexpensively re-provision technological infrastructure resources. The cost of overall computing is unchanged, however, and the providers will merely absorb up-front costs and spread costs over a longer period.[38]
- **Cost** is claimed to be greatly reduced and capital expenditure is converted to operational expenditure[39]. This ostensibly lowers barriers to entry, as infrastructure is typically provided by a third-party and does not need to be purchased for one-time or infrequent intensive computing tasks. Pricing on a utility computing basis is fine-grained with usage-based options and fewer IT skills are required for implementation (in-house).[40] Some would argue that given the low cost of computing resources, that the IT burden merely shifts the cost from in-house to outsourced providers. Furthermore, any cost reduction benefit must be weighed against a corresponding loss of control, access and security risks.
- **Device and location independence**[41] enable users to access systems using a web browser regardless of their location or what device they are using (e.g., PC, mobile). As infrastructure is off-site (typically provided by a third-party) and accessed via the Internet, users can connect from anywhere.[40]
- **Multi-tenancy** enables sharing of resources and costs across a large pool of

users thus allowing for:
  - **Centralization** of infrastructure in locations with lower costs (such as real estate, electricity, etc.)
  - **Peak-load capacity** increases (users need not engineer for highest possible load-levels)
  - **Utilization and efficiency** improvements for systems that are often only 10–20% utilized.[27]
- **Reliability** improves through the use of multiple redundant sites, which makes cloud computing suitable for business continuity and disaster recovery.[42] Nonetheless, many major cloud computing services have suffered outages, and IT and business managers can at times do little when they are affected.[43][44]
- **Scalability** via dynamic ("on-demand") provisioning of resources on a fine-grained, self-service basis near real-time, without users having to engineer for peak loads. Performance is monitored, and consistent and loosely-coupled architectures are constructed using web services as the system interface.[40]
- **Security** typically improves due to centralization of data[45], increased security-focused resources, etc., but concerns can persist about loss of control over certain sensitive data. Security is often as good as or better than under traditional systems, in part because providers are able to devote resources to solving security issues that many customers cannot afford[46]. Providers typically log accesses, but accessing the audit logs themselves can be difficult or impossible. Ownership, control and access to data controlled by "cloud" providers may be made more difficult,just as it is sometimes difficult to gain access to "live" support with current utilities. Under the cloud paradigm, management of sensitive data is placed in the hands of cloud providers and third parties. Currently, many developers are implementing OAuth (open protocol for secure API authorization), as it allows more granularity of data controls across cloud applications. OAuth is an open protocol, initiated by Blain Cook and Chris Messina, to allow secure API authorization in a standard method for desktop, mobile, and web applications.
- **Sustainability** comes about through improved resource utilization, more efficient systems, and carbon neutrality.[47][48] Nonetheless, computers and associated infrastructure are major consumers of energy. A given (server-based) computing task will use X amount of energy whether it is on-site, or off. [49]

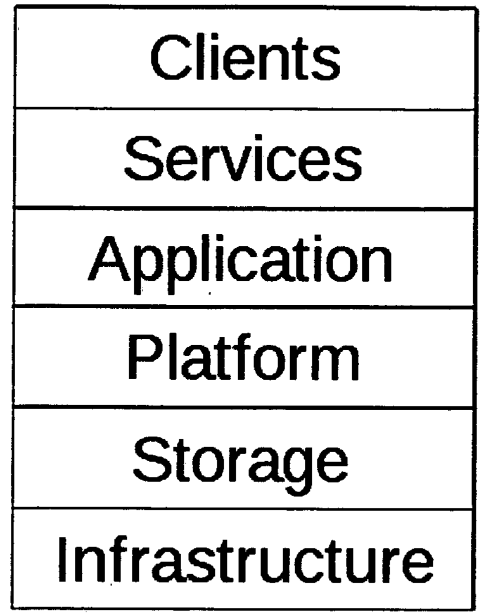
Components

Application

See also category: Cloud applications

A *cloud application* leverages the Cloud in software architecture, often eliminating the need to install and run the application on the customer's own computer, thus alleviating the burden of software maintenance, ongoing operation, and support. For example:

- Peer-to-peer / volunteer computing (Bittorrent, BOINC Projects, Skype)
- Web application (Twitter)
- Software as a service (Google Apps, SAP and Salesforce)
- Software plus services (Microsoft Online Services)



Six layers components of cloud computing

Client

See also category: Cloud clients

A *cloud client* consists of computer hardware and/or computer software which relies on cloud computing for application delivery, or which is specifically designed for delivery of cloud services and which, in either case, is essentially useless without it.[50] For example:

- Mobile (Android, iPhone, Windows Mobile)[51][52][53]
- Thin client (CherryPal, Zonbu, gOS-based systems)[54][55][56]
- Thick client / Web browser (Microsoft Internet Explorer, Mozilla Firefox)

Infrastructure

See also category: Cloud infrastructure

Cloud infrastructure, such as Infrastructure as a service, is the delivery of computer infrastructure, typically a platform virtualization environment, as a service.[57] For example:

- Full virtualization (GoGrid, Skytap, iland)
- Grid computing (Sun Cloud)
- Hosted desktop (ThinkGrid[58])
- Management (RightScale)
- Compute (Amazon Elastic Compute Cloud)
- Platform (Force.com)
- Storage (Amazon S3, Nirvanix, Rackspace)

Platform

See also category: Cloud platforms

A *cloud platform*, such as Platform as a service, the delivery of a computing platform, and/or solution stack as a service, facilitates deployment of applications without the cost and complexity of buying and managing the underlying hardware and software layers.[59] For example:

- Code Based Web Application Frameworks
  - Java Google Web Toolkit (Google App Engine)
  - Python Django (Google App Engine)
  - Ruby on Rails (Heroku)
  - .NET (Azure Services Platform)
- Non-Code Based Web Application Framework
  - WorkXpress
- Cloud Computing Application & Web Hosting (Rackspace Cloud)
- Proprietary (Force.com)

Service

See also category: Cloud services

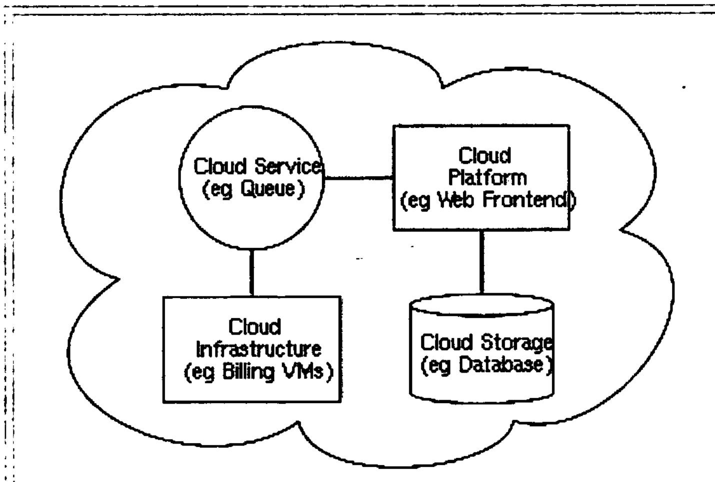
A *cloud service* includes "products, services and solutions that are delivered and consumed in real-time over the Internet"[40] For example, Web Services ("software

system[s] designed to support interoperable machine-to-machine interaction over a network")[60] which may be accessed by other cloud computing components, software, e.g., Software plus services, or end users directly.[61] Specific examples include:

- Identity (OAuth, OpenID)
- Integration (Amazon Simple Queue Service)
- Payments (Amazon Flexible Payments Service, Google Checkout, PayPal)
- Mapping (Google Maps, Yahoo! Maps, MapQuest)
- Search (Alexa, Google Custom Search, Yahoo! BOSS)
- Video Games (OnLive, Gaikai)
- Live chat (LivePerson)
- Symplified Symplified Inc. is a privately owned American company based in Boulder, CO. Symplified was founded by the same management team that created Securant, which pioneered the market for Web access management software and was acquired for $140M by RSA Security. The company first incorporated in 2006.
- Others (Amazon Mechanical Turk)

Architecture

Cloud architecture,[62] the systems architecture of the software systems involved in the delivery of *cloud computing*, comprises hardware and software designed by a *cloud architect* who typically works for a *cloud integrator*. It typically involves multiple *cloud components* communicating with each other over application programming interfaces, usually web services.[63]



Cloud computing sample architecture

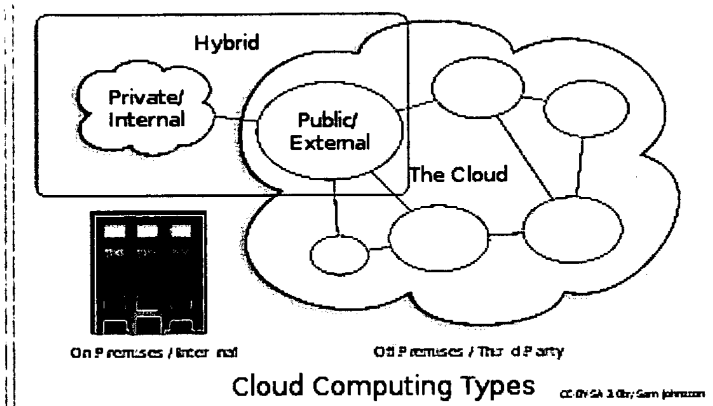
This closely resembles the Unix philosophy of having multiple programs doing one thing well and working together over universal interfaces. Complexity is controlled and the resulting systems are more manageable than their monolithic counterparts.

Cloud architecture extends to the client, where web browsers and/or software applications access *cloud applications*.

Cloud storage architecture is loosely coupled, where metadata operations are centralized enabling the data nodes to scale into the hundreds, each independently delivering data to applications or users.

Types

Public cloud



Cloud computing types

Public cloud or *external cloud* describes cloud computing in the traditional mainstream sense, whereby resources are dynamically provisioned on a fine-grained, self-service basis over the Internet, via web applications/web services, from an off-site third-party provider who shares resources and bills on a fine-grained utility computing basis.[40]

Hybrid cloud

A *hybrid cloud* environment consisting of multiple internal and/or external providers[64] "will be typical for most enterprises".[65]

Private cloud

Private cloud and *internal cloud* are neologisms that some vendors have recently used to describe offerings that emulate cloud computing on private networks. These (typically virtualisation automation) products claim to "deliver some benefits of cloud computing without the pitfalls", capitalising on data security, corporate governance, and reliability concerns. They have been criticized on the basis that users "still have to buy, build, and manage them" and as such do not benefit from lower up-front capital costs and less hands-on management[65], essentially

"[lacking] the economic model that makes cloud computing such an intriguing concept".[66][67]

While an analyst predicted in 2008 that private cloud networks would be the future of corporate IT,[68] there is some uncertainty whether they are a reality even within the same firm.[69] Analysts also claim that within five years a "huge percentage" of small and medium enterprises will get most of their computing resources from external cloud computing providers as they "will not have economies of scale to make it worth staying in the IT business" or be able to afford private clouds.[70]. Analysts have reported on Platform's view that private clouds are a stepping stone to external clouds, particularly for the financial services, and that future datacenters will look like internal clouds. [71]

The term has also been used in the logical rather than physical sense, for example in reference to platform as a service offerings[72], though such offerings including Microsoft's Azure Services Platform are not available for on-premises deployment. [73]

Roles

Provider

See also category: Cloud computing providers

A *cloud computing provider* or *cloud computing service provider* owns and operates live *cloud computing* systems to deliver service to third parties. Usually this requires significant resources and expertise in building and managing next-generation data centers. Some organisations realise a subset of the benefits of cloud computing by becoming "internal" cloud providers and servicing themselves, although they do not benefit from the same economies of scale and still have to engineer for peak loads. The barrier to entry is also significantly higher with capital expenditure required and billing and management creates some overhead. Nonetheless, significant operational efficiency and agility advantages can be realised, even by small organisations, and server consolidation and virtualization rollouts are already well underway.[74] Amazon.com was the first such provider, modernising its data centers which, like most computer networks, were using as little as 10% of its capacity at any one time just to leave room for occasional spikes.

This allowed small, fast-moving groups to add new features faster and easier, and they went on to open it up to outsiders as Amazon Web Services in 2002 on a utility computing basis.[27]

The companies listed in the *Components* section are providers.

User

See also category: Cloud computing users

A user is a consumer of *cloud computing*.[50] The privacy of users in cloud computing has become of increasing concern.[75] The rights of users is also an issue, which is being addressed via a community effort to create a bill of rights.[76][77][78] The Franklin Street statement was drafted with an eye towards protecting users' freedoms.[79]

Vendor

See also category: Cloud computing vendors

Some vendors sell or give products and services that facilitate the delivery, adoption and use of *cloud computing*.[80] For example:

- Computer hardware (Dell, HP, IBM, and Sun Microsystems)
  - Storage (3PAR, EMC, Hitachi Data Systems, IBM, Mezeo, NetApp, ParaScale, and Sun Microsystems)
  - Infrastructure (Cisco Systems, Juniper Networks, and Brocade Communications)
- Computer software (3tera, Eucalyptus, g-Eclipse, and Hadoop)
  - Operating systems (Solaris, AIX, Linux including Novell[81], Red Hat[82], and Ubuntu[83])
  - Platform virtualization (Citrix, IBM, Linux KVM, Microsoft, Sun xVM, VMware, and Xen)

Standards

See also category: Cloud standards

Cloud standards, a number of existing, typically lightweight, open standards, have facilitated the growth of cloud computing, including:[84]

- **Application**
  - Communications (HTTP, XMPP)
  - Security (OAuth, OpenID, SSL/TLS[85])
  - Syndication (Atom)
- **Client**
  - Browsers (AJAX)
  - Offline (HTML 5)
- **Implementations**
  - Virtualization (OVF[86])
- **Platform**
  - Solution stacks (LAMP)
- **Service**
  - Data (XML, JSON)
  - Web Services (REST)
- **Storage**

See also

- Cloud Computing Manifesto
- Cloud networking

References

1. ^ "Gartner Says Cloud Computing Will Be As Influential As E-business (http://www.gartner.com/it/page.jsp?id=707508)". *www.gartner.com*. Gartner. 2008-06-26. http://www.gartner.com/it/page.jsp?id=707508. Retrieved 2009-06-02.
2. ^ Gruman, Galen (2008-04-07). "What cloud computing really means (http://www.infoworld.com/article/08/04/07/15FE-cloud-computing-reality_1.html)". *InfoWorld*. http://www.infoworld.com/article/08/04/07/15FE-cloud-computing-reality_1.html. Retrieved 2009-06-02.
3. ^ Distinguishing Cloud Computing from Utility Computing (http://www.ebizq.net/blogs/saasweek/2008/03/distinguishing_cloud_computing/)
4. ^ Williams, John M.; Chris Sears (http://twitter.com/csears) (2008-12-31). "Who Coined the Phrase Cloud Computing? (http://www.johnmwillis.com/cloud-

computing/who-coined-the-phrase-cloud-computing/)". http://www.johnmwillis.com/cloud-computing/who-coined-the-phrase-cloud-computing/. Retrieved 2009-03-03.

5. ^ Anita Campbell (http://www.theappgap.com/author/acampbell/) (2008-08-31). "Cloud Computing - Get Used to the Term (http://www.theappgap.com/cloud-computing-get-used-to-the-term.html)". The App Gap. http://www.theappgap.com/cloud-computing-get-used-to-the-term.html. Retrieved 2009-03-03.
6. ^ *a b* The Internet Cloud (http://www.thestandard.com/article/0,1902,5466,00.html)
7. ^ "Cloud computing---emerging paradigm for computing (http://meetings.informs.org/Dallas97/TALKS/C20.html)". http://meetings.informs.org/Dallas97/TALKS/C20.html.
8. ^ "It's probable that you've misunderstood 'Cloud Computing' until now (http://portal.acm.org/citation.cfm?id=1496091.1496100&coll=&dl=ACM&CFID=21518680&CFTOKEN=18800807)". TechPluto. http://portal.acm.org/citation.cfm?id=1496091.1496100&coll=&dl=ACM&CFID=21518680&CFTOKEN=18800807.
9. ^ What's In A Name? Utility vs. Cloud vs Grid (http://www.datacenterknowledge.com/archives/2008/Mar/25/whats_in_a_name_utility_
10. ^ I.B.M. to Push 'Cloud Computing,' Using Data From Afar (http://www.nytimes.com/2007/11/15/technology/15blue.html)
11. ^ Overheard: What the heck is computing in a cloud? (http://itknowledgeexchange.techtarget.com/overheard/overheard-what-the-heck-is-computing-in-a-cloud/)
12. ^ ACM Ubiquity: Emergence of The Academic Computing Cloud (http://www.acm.org/ubiquity/volume_9/v9i31_delic.html)
13. ^ [Katarina Stanoevska-Slabeva, Davide Maria Parrilli, George A. Thanos: BEinGRID: Development of Business Models for the Grid Industry. GECON 2008: 140-151]
14. ^ Grid Voices blog (http://www.gridipedia.eu/grid_voices/?p=57)
15. ^ Cloud Computing: The Evolution of Software-as-a-Service (http://knowledge.wpcarey.asu.edu/article.cfm?articleid=1614)
16. ^ Forrester's Advice to CFOs: Embrace Cloud Computing to Cut Costs (http://www.eweek.com/c/a/Enterprise-Applications/Forresters-Advice-to-CFOs-Embrace-Cloud-Computing-to-Cut-Costs/)
17. ^ Five cloud computing questions (http://www.networkworld.com/columnists/2008/080508-dzubeck.html)
18. ^ Nicholas Carr on 'The Big Switch' to cloud computing (http://www.computerworlduk.com/technology/internet/applications/instant-expert/index.cfm?articleid=1610)
19. ^ 1 Midsize Organization Busts 5 Cloud Computing Myths (http://www.bmighty.com/services/showArticle.jhtml?articleID=211600030)
20. ^ Cloud Computing Savings - Real or Imaginary? (http://www.appirio.com/blog/2009/04/cloud-computing-savings-real-or.php)
21. ^ Google Apps makes its way into big business

(http://www.computerweekly.com/Articles/2008/06/24/231178/google-apps-makes-its-way-into-big-business.htm)
22. ^ Google, Inc. Q2 2008 Earnings Call (http://seekingalpha.com/article/85608-google-inc-q2-2008-earnings-call-transcript)
23. ^ Buyya, Rajkumar; Chee Shin Yeo, Srikumar Venugopal (PDF). *Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities (http://www.gridbus.org/~raj/papers/hpcc2008_keynote_cloudcomputing.pdf)*. Department of Computer Science and Software Engineering, The University of Melbourne, Australia. pp. 9. http://www.gridbus.org/~raj/papers/hpcc2008_keynote_cloudcomputing.pdf. Retrieved 2008-07-31.
24. ^ Open source fuels growth of cloud computing, software-as-a-service (http://www.networkworld.com/news/2008/072808-open-source-cloud-computing.html)
25. ^ July, 1993 meeting report from the IP over ATM working group of the IETF (http://mirror.switch.ch/ftp/doc/ietf/ipatm/atm-minutes-93jul.txt)
26. ^ Internet Critic Takes on Microsoft (http://www.nytimes.com/2001/04/09/technology/09HAIL.html?ex=1217563200&en=7c46bdefb6a8450a&ei=5070)
27. ^ ***a b c*** Jeff Bezos' Risky Bet (http://www.businessweek.com/magazine/content/06_46/b4009001.htm)
28. ^ Google and I.B.M. Join in 'Cloud Computing' Research (http://www.nytimes.com/2007/10/08/technology/08cloud.html?_r=1&ex=1349496000&en=92627f0f65ea0d75&ei=5090&partner=rssuserland&emc=rs
29. ^ Keep an eye on cloud computing (http://www.networkworld.com/newsletters/itlead/2008/070708itlead1.html)
30. ^ Gartner Says Worldwide IT Spending On Pace to Surpass $3.4 Trillion in 2008 (http://www.gartner.com/it/page.jsp?id=742913)
31. ^ The Top Cloud Computing Solutions people are looking for in 2009– Survey by onCloudComputing.com (http://www.oncloudcomputing.com/en/2009/07/the-top-cloud-computing-solutions-people-are-looking-for-in-2009â? ? -survey-by-oncloudcomputingcom/)
32. ^ Cloud computing: Are there dangers to having information infrastructure, software and services hosted on the internet rather than on our own personal computers? (http://technology.timesonline.co.uk/tol/news/tech_and_web/article3874599.ece)
33. ^ Cloud computing is a trap, warns GNU founder Richard Stallman, Guardian, September 30, 2008 (http://www.guardian.co.uk/technology/2008/sep/29/cloud.computing.richard.stallman/)
34. ^ ***a b*** Computers without borders (http://www.economist.com/specialreports/displaystory.cfm?story_id=12411854)
35. ^ Feature Guide: Amazon EC2 Availability Zones (http://developer.amazonwebservices.com/connect/entry.jspa?externalID=1347&categoryID=112)
36. ^ Microsoft Plans 'Cloud' Operating System (http://www.nytimes.com/2008/10/28/technology/28soft.html)

37. ^ Brodkin, Jon (July 02, 2008). "Gartner: Seven cloud-computing security risks (http://www.infoworld.com/d/security-central/gartner-seven-cloud-computing-security-risks-853)". *www.infoworld.com* (infoworld). http://www.infoworld.com/d/security-central/gartner-seven-cloud-computing-security-risks-853. Retrieved 2009-04-15.
38. ^ Infrastructure Agility: Cloud Computing as a Best Practice (http://edgewatertech.wordpress.com/2009/04/24/best-practice-cloud-computing/)
39. ^ Recession Is Good For Cloud Computing – Microsoft Agrees (http://www.cloudave.com/link/recession-is-good-for-cloud-computing-microsoft-agrees)
40. ^ *a b c d e* Defining "Cloud Services" and "Cloud Computing" (http://blogs.idc.com/ie/?p=190)
41. ^ The new geek chic: Data centers (http://news.cnet.com/8301-13953_3-9977049-80.html)
42. ^ Cloud Computing: Small Companies Take Flight (http://www.data-storage-today.com/news/Cloud-Computing--Firms-Take-Flight/story.xhtml?story_id=0130012A6383)
43. ^ Google Apps Admins Jittery About Gmail, Hopeful About Future (http://www.pcworld.com/businesscenter/article/149892/google_apps_admins_jittery_at
44. ^ New Resource, Born of a Cloud Feud (http://www.datacenterknowledge.com/archives/2008/09/22/a-new-resource-born-of-a-cloud-feud/A)
45. ^ Exari: Death By Laptop (http://exari.blogspot.com/2006/05/death-by-laptop.html)
46. ^ Cloud computing security forecast: Clear skies (http://news.cnet.com/8301-1009_3-10150569-83.html)
47. ^ Google to go carbon neutral by 2008 (http://news.zdnet.co.uk/internet/0,1000000097,39287634,00.htm)
48. ^ What is Cloud Computing? (http://jobsearchtech.about.com/od/historyoftechindustry/a/cloud_computing.htm)
49. ^ Shut off your computer (http://www.time.com/time/specials/2007/environment/article/0,28804,1602354_160307
50. ^ *a b* Nimbus Cloud Guide (http://workspace.globus.org/vm/TP2.1/doc/cloud.html)
51. ^ Google's Open Source Android OS Will Free the Wireless Web (http://www.wired.com/techbiz/media/magazine/16-07/ff_android?currentPage=all)
52. ^ In Sync to Pierce the Cloud (http://www.nytimes.com/2008/07/17/technology/personaltech/17pogue.html?_r=1&oref=slogin)
53. ^ Microsoft demos mobile cloud sync client (http://www.windowsfordevices.com/news/NS6388605875.html)
54. ^ CherryPal brings cloud computing to the masses (http://www.itbusinessedge.com/cm/community/news/inf/blog/cherrypal-brings-cloud-computing-to-the-masses/?cs=19874)
55. ^ Zonbu has alluring features, price (http://www.newsobserver.com/business/technology/gilster/2007/story/771924.html)
56. ^ GOS cloud computing (http://www.businesswire.com/portal/site/home/index.jsp?

epi_menuItemID=887566059a3aedb6efaaa9e27a808a0c&ndmViewId=news_view&ndn

57. ^ EMC buys Pi and forms a cloud computing group (http://searchstorage.techtarget.com/news/article/0,289142,sid5_gci1301852,00.html)
58. ^ http://www.thinkgrid.co.uk/
59. ^ Google angles for business users with 'platform as a service' (http://www.guardian.co.uk/technology/2008/apr/17/google.software)
60. ^ "Web Services Glossary (http://www.w3.org/TR/ws-gloss/)". http://www.w3.org/TR/ws-gloss/.
61. ^ The Emerging Cloud Service Architecture (http://aws.typepad.com/aws/2008/06/the-forthcoming.html)
62. ^ Building GrepTheWeb in the Cloud, Part 1: Cloud Architectures (http://developer.amazonwebservices.com/connect/entry.jspa?externalID=1632&categoryID=100)
63. ^ Cloud Maturity Is Accelerating: More Than Just Reaction To The Hype? (http://blogs.forrester.com/it_infrastructure/2008/07/cloud-maturity.html)
64. ^ IBM Embraces Juniper For Its Smart 'Hybrid Cloud', Disses Cisco (IBM) (http://www.businessinsider.com/2009/2/ibm-embraces-juniper-for-its-smart-hybrid-cloud-disses-cisco-ibm)
65. ^ *a* *b* Private Clouds Take Shape (http://www.informationweek.com/news/services/business/showArticle.jhtml?articleID=209904474)
66. ^ Just don't call them private clouds (http://news.cnet.com/8301-13556_3-10150841-61.html)
67. ^ There's No Such Thing As A Private Cloud (http://www.informationweek.com/cloud-computing/blog/archives/2009/01/theres_no_such.html)
68. ^ Private cloud networks are the future of corporate IT (http://www.networkworld.com/news/2008/111208-private-cloud-networks.html)
69. ^ Private Cloud Computing: The Only Thing Real so Far is the Desire (http://blogs.gartner.com/nick_gall/2009/02/09/private-cloud-computing-the-only-thing-real-so-far-is-the-desire/)
70. ^ Million-Dollar Private Clouds (http://www.informationweek.com/cloud-computing/blog/archives/2009/01/milliondollar_p.html)
71. ^ From Grid to Cloud (Gridipedia) (http://www.gridipedia.eu/grid-to-cloud.html)
72. ^ Google opens private cloud to coders (http://www.theregister.co.uk/2008/04/08/google_unveils_app_engine/)
73. ^ Microsoft Nixes Private Azure Clouds (http://www.informationweek.com/news/windows/operatingsystems/showArticle.jhtml?articleID=216300168&cid=nl_IWK_daily_H)
74. ^ ACM Queue - Beyond Server Consolidation (http://www.acmqueue.org/modules.php?name=Content&pa=showpage&pid=528)
75. ^ Google Privacy Practices Worse Than ISP Snooping, AT&T Charges (http://blog.wired.com/27bstroke6/2008/08/google-privacy.html)
76. ^ The bill of rights is currently in draft (http://wiki.cloudcommunity.org/wiki/CloudComputing:Bill_of_Rights).

77. ^ Draft Cloud Computing: Bill of Rights Now Available (http://saas.sys-con.com/read/filter/saaslatest.htm)
78. ^ Johnston, Sam; Urquhart, James; Wellner, Rich (2008-09-16). "Cloud Computing:Bill of rights (http://wiki.cloudcommunity.org/wiki/CloudComputing:Bill_of_Rights)". http://wiki.cloudcommunity.org/wiki/CloudComputing:Bill_of_Rights. Retrieved 2008-09-16.
79. ^ http://autonomo.us/2008/07/franklin-street-statement/
80. ^ List of Cloud Platforms, Providers and Enablers (http://groups.google.ca/group/cloud-computing/web/list-of-cloud-platforms-providers-and-enablers)
81. ^ Jeff Jaffe, Software Appliances and Cloud Computing (http://www.novell.com/ctoblog/?p=119)
82. ^ Red Hat chief: 'The clouds will all run Linux' (http://resources.zdnet.co.uk/articles/features/0,1000002000,39454819,00.htm)
83. ^ Ubuntu 9.04 beta out, now with fresh Eucalyptus (http://mobile.pcauthority.com.au/Article.aspx?CIID=141212&type=News&page=0&showall=true)
84. ^ The Cloud and Standards (http://redmonk.com/sogrady/2008/06/24/cloud_standards/)
85. ^ Lock-in, security loom as dark side of Compute Cloud (http://searchsoa.techtarget.com/news/article/0,289142,sid26_gci1328293,00.html)
86. ^ LinuxWorld/Next Generation Data Center attendees get schooled in cloud computing (http://searchdatacenter.techtarget.com/news/article/0,289142,sid80_gci1324347,00.htm

External links

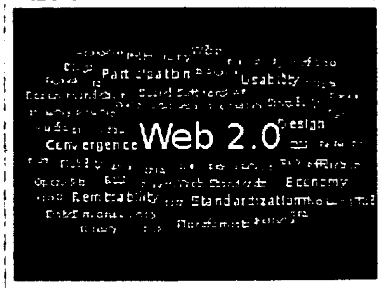
- Luis M. Vaquero et al., A Break in the Clouds: Toward a Cloud Definition (http://portal.acm.org/citation.cfm?id=1496091.1496100), ACM SIGCOMM Computer Communication Review, Volume 39, Issue 1 (January 2009), Pages 50–55, ISSN:0146-4833
- Jones, M. Tim, *Cloud Computing with Linux* (http://www.ibm.com/developerworks/linux/library/l-cloud-computing/) from IBM DeveloperWorks (2008-09-10).
- Chappell, David (August 2008) (PDF). *A Short Introduction to Cloud Platforms (http://www.davidchappell.com/CloudPlatforms--Chappell.pdf)*. David Chappell & Associates. http://www.davidchappell.com/CloudPlatforms--Chappell.pdf. Retrieved 2008-08-20.
- Social Directory of SaaS Cloud Computing Companies (http://www.theeasybee.com/directory/saas-cloud-computing-and-other/)
- What is Cloud Computing ? - Web 2.0 expo (http://videos.techielife.com/what-is-cloud-computing/video-online/2008/11/13) - A video where Tim O'Reilly, Dan Farber, Matt Mullenweg and others answer this question.

Retrieved from "http://en.wikipedia.org/wiki/Cloud_computing"

Categories: Cloud computing | Distributed data sharing
Hidden categories: Wikipedia pages move-protected due to vandalism | Articles containing potentially dated statements from 2009 | All articles containing potentially dated statements | All articles with unsourced statements | Articles with unsourced statements from August 2009 | Articles with unsourced statements from April 2009

Web 2.0

From Wikipedia, the free encyclopedia



A tag cloud (a typical Web 2.0 phenomenon in itself) presenting Web 2.0 themes

"**Web 2.0**" is commonly associated with web development and web design that facilitates interactive information sharing, interoperability, user-centered design[1] and collaboration on the World Wide Web. Examples of Web 2.0 include web-based communities, hosted services, web applications, social-networking sites, video-sharing sites, wikis, blogs, mashups and folksonomies. A Web 2.0 site allows its users to interact with other users or to change website content, in contrast to non-interactive websites where users are limited to the passive viewing of information that is provided to them.

The term is closely associated with Tim O'Reilly because of the O'Reilly Media Web 2.0 conference in 2004.[2][3] Although the term suggests a new version of the World Wide Web, it does not refer to an update to any technical specifications, but rather to cumulative changes in the ways software developers and end-users use the Web. Whether Web 2.0 is qualitatively different from prior web technologies has been challenged by World Wide Web inventor Tim Berners-Lee who called the term a "piece of jargon"[4].

Contents

History

The term "Web 2.0" was coined by Darcy DiNucci in 1999. In her article "Fragmented Future," she writes[5]

> The Web we know now, which loads into a browser window in essentially static screenfuls, is only an embryo of the Web to come. The first glimmerings of Web 2.0 are beginning to appear, and we are just starting to see how that embryo might develop The Web will be understood not as screenfuls of text and graphics but as a transport mechanism, the ether through which interactivity happens. It will [...] appear on your computer screen, [...] on your TV set [...] your car dashboard [...] your cell phone [...] hand-held game machines [...] and maybe even your microwave.

Her use of the term deals mainly with Web design and aesthetics; she argues that the Web is "fragmenting" due

to the widespread use of portable Web-ready devices. Her article is aimed at designers, reminding them to code for an ever-increasing variety of hardware. As such, her use of the term hints at - but does not directly relate to - the current uses of the term.

The term did not resurface until 2003.[6][7][8] These authors focus on the concepts currently associated with the term where, as Scott Dietzen puts it, "the Web becomes a universal, standards-based integration platform."[9]

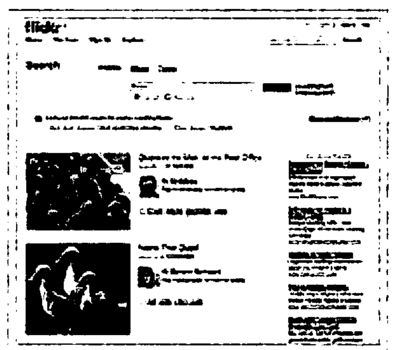
In 2004, the term began its rise in popularity when O'Reilly Media and MediaLive hosted the first Web 2.0 conference. In their opening remarks, John Batelle and Tim O'Reilly outlined their definition of the "Web as Platform," where software applications are built upon the Web as opposed to upon the desktop. The unique aspect of this migration, they argued, is that "customers are building your business for you."[10] They argued that the activities of users generating content (in the form of ideas, text, videos, or pictures) could be "harnessed" to create value. According to Tim O'Reilly:

> Web 2.0 is the business revolution in the computer industry caused by the move to the Internet as a platform, and an attempt to understand the rules for success on that new platform.[11]

From there, the term Web 2.0 was largely championed by bloggers and by technology journalists, culminating in the 2006 TIME magazine Person of The Year - "You."[12] That is, TIME selected the masses of users who were participating in content creation on social networks, blogs, wikis, and media sharing sites. The cover story author Lev Grossman explains:

> It's a story about community and collaboration on a scale never seen before. It's about the cosmic compendium of knowledge Wikipedia and the million-channel people's network YouTube and the online metropolis MySpace. It's about the many wresting power from the few and helping one another for nothing and how that will not only change the world, but also change the way the world changes.

Characteristics



Flickr, a Web 2.0 web site that allows users to upload and share photos

Web 2.0 websites allow users to do more than just retrieve information. They can build on the interactive facilities of "Web 1.0" to provide "Network as platform" computing, allowing users to run software-applications entirely through a browser.[3] Users can own the data on a Web 2.0 site and exercise control over that data.[3][13] These sites may have an "Architecture of participation" that encourages users to add value to the application as they use it.[2][3] This stands in contrast to traditional websites, the sort that limited visitors to viewing and whose content only the site's owner could modify. Web 2.0 sites often feature a rich, user-friendly interface based on Ajax[2][3] and similar client-side interactivity frameworks, or full client-server application frameworks such as OpenLaszlo, Flex, and the ZK framework.[3][13].

The concept of Web-as-participation-platform captures many of these characteristics. Bart Decrem, a founder and former CEO of Flock, calls Web 2.0 the "participatory Web"[14] and regards the Web-as-information-source as Web 1.0.

The impossibility of excluding group-members who don't contribute to the provision of goods from sharing profits gives rise to the possibility that rational members will prefer to withhold their contribution of effort and free-ride on the contribution of others.[15] This requires what is sometimes called Radical Trust by the management of the website. According to Best,[16] the characteristics of Web 2.0 are: rich user experience, user

participation, dynamic content, metadata, web standards and scalability. Further characteristics, such as openness, freedom[17] and collective intelligence[18] by way of user participation, can also be viewed as essential attributes of Web 2.0.

Technology overview

Web 2.0 draws together the capabilities of client- and server-side software, content syndication and the use of network protocols. Standards-oriented web browsers may use plugins and software extensions to handle the content and the user interactions. Web 2.0 sites provide users with information storage, creation, and dissemination capabilities that were not possible in the environment now known as "Web 1.0".

Web 2.0 websites typically include some of the following features and techniques. Andrew McAfee used the acronym SLATES to refer to them:[19]

Search
: Finding information through keyword search.

Links
: Guides to other related information.

Authoring
: The ability to create and update content leads to the collaborative work of many rather than just a few web authors. In wikis, users may extend, undo and redo each other's work. In blogs, posts and the comments of individuals build up over time.

Tags
: Categorization of content by users adding one-word descriptions to facilitate searching, without dependence on pre-made categories.

Extensions
: Software that makes the Web an application platform as well as a document server.

Signals
: The use of syndication technology such as RSS to notify users of content changes.

How it works

The client-side/web browser technologies typically used in Web 2.0 development are Asynchronous JavaScript and XML (Ajax), Adobe Flash, and JavaScript/Ajax frameworks such as Yahoo! UI Library, Dojo Toolkit, MooTools, and jQuery. Ajax programming uses JavaScript to upload and download new data from the web server without undergoing a full page reload.

The data fetched by an Ajax request is typically formatted in XML or JSON (JavaScript Object Notation) format, two widely used structured data formats. Since both of these formats are natively understood by JavaScript, a programmer can easily use them to transmit structured data in their web application. When this data is received via Ajax, the JavaScript program then uses the Document Object Model (DOM) to dynamically update the web page based on the new data, allowing for a rapid and interactive user experience.

Adobe Flash is another technology often used in Web 2.0 applications. As a widely available plugin independent of W3C (World Web Consortium, the governing body of web standards and protocols), standards, Flash is capable of doing many things which are not currently possible in HTML, the language used to construct web pages. Of Flash's many capabilities, the most commonly used in Web 2.0 is its ability to play audio and video files. This fact alone has allowed for the creation of cutting edge Web 2.0 sites such as YouTube, where rich media is gracefully integrated with standard HTML.

In addition to Flash and Ajax, JavaScript/Ajax frameworks have recently become a very popular means of

creating Web 2.0 sites. At their core, these frameworks do not use technology any different from JavaScript, Ajax, and the DOM. What frameworks do is smooth over inconsistencies between web browsers and extend the functionality available to developers. Many of them also come with customizable, prefabricated 'widgets' that accomplish such common tasks as picking a date from a calendar, displaying a data chart, making a tabbed panel, etc.

On the server side, Web 2.0 uses many of the same technologies as Web 1.0. Languages such as PHP, Ruby, Perl, Python, and ASP are used by developers to dynamically output data using information from files and databases. What has begun to change in Web 2.0 is the way this data is formatted. In the early days of the internet, there was little need for different websites to communicate with each other and share data. In the new 'participatory web', however, sharing data between sites has become an essential capability. To share its data with other sites, a web site must be able to generate output in machine-readable formats such as XML, RSS, and JSON. When a site's data is available in one of these formats, another website can use it to integrate a portion of that site's functionality into itself, linking the two together. When this design pattern is implemented, it ultimately leads to data that is both easier to find and more thoroughly categorized, a hallmark of the philosophy behind the Web 2.0 movement.

Usage

The popularity of the term Web 2.0, along with the increasing use of blogs, wikis, and social networking technologies, has led many in academia and business to coin a flurry of 2.0s,[20] including Library 2.0,[21] Social Work 2.0,[22] Enterprise 2.0, PR 2.0,[23] Classroom 2.0, Publishing 2.0, Medicine 2.0, Travel 2.0 and Government 2.0.[24] Many of these 2.0s refer to Web 2.0 technologies as the source of the new version in their respective disciplines and areas. For example, in the Talis white paper "Library 2.0: The Challenge of Disruptive Innovation," Paul Miller argues

> Blogs, wikis and RSS are often held up as exemplary manifestations of Web 2.0. A reader of a blog or a wiki is provided with tools to add a comment or even, in the case of the wiki, to edit the content. This is what we call the Read/Write web.Talis believes that Library 2.0 means harnessing this type of participation so that libraries can benefit from increasingly rich collaborative cataloguing efforts, such as including contributions from partner libraries as well as adding rich enhancements, such as book jackets or movie files, to records from publishers and others.[25]

Here, Miller links Web 2.0 technologies and the culture of participation that they engender to the field of library science, supporting his claim that there is now a "Library 2.0." Many of the other proponents of new 2.0s mentioned here use similar methods.

According to the Global Language Monitor, Web 2.0 is the one millionth word to enter the English language [26].

Web-based applications and desktops

Ajax has prompted the development of websites that mimic desktop applications, such as word processing, the spreadsheet, and slide-show presentation. WYSIWYG wiki sites replicate many features of PC authoring applications. Still other sites perform collaboration and project management functions. In 2006 Google, Inc. acquired one of the best-known sites of this broad class, Writely.[27]

Several browser-based "operating systems" have emerged, including EyeOS[28] and YouOS.[29] Although coined as such, many of these services function less like a traditional operating system and more as an application platform. They mimic the user experience of desktop operating-systems, offering features and applications similar to a PC environment, as well as the added ability of being able to run within any modern

browser. However, these operating systems do not control the hardware on the client's computer.

Numerous web-based application services appeared during the dot-com bubble of 1997–2001 and then vanished, having failed to gain a critical mass of customers. In 2005, WebEx acquired one of the better-known of these, Intranets.com, for $45 million.[30]

Internet applications

Main article: Rich Internet application

XML and RSS

Advocates of "Web 2.0" may regard syndication of site content as a Web 2.0 feature, involving as it does standardized protocols, which permit end-users to make use of a site's data in another context (such as another website, a browser plugin, or a separate desktop application). Protocols which permit syndication include RSS (Really Simple Syndication — also known as "web syndication"), RDF (as in RSS 1.1), and Atom, all of them XML-based formats. Observers have started to refer to these technologies as "Web feed" as the usability of Web 2.0 evolves and the more user-friendly Feeds icon supplants the RSS icon.

Specialized protocols

Specialized protocols such as FOAF and XFN (both for social networking) extend the functionality of sites or permit end-users to interact without centralized websites.

Other protocols, like XMPP enables services to users like Services over the Messenger

Web APIs

Machine-based interaction, a common feature of Web 2.0 sites, uses two main approaches to web APIs, which allow web-based access to data and functions: REST and SOAP.

1. REST (Representational State Transfer) web APIs use HTTP alone to interact, with XML (eXtensible Markup Language) or JSON payloads;
2. SOAP involves POSTing more elaborate XML messages and requests to a server that may contain quite complex, but pre-defined, instructions for the server to follow.

Often servers use proprietary APIs, but standard APIs (for example, for posting to a blog or notifying a blog update) have also come into wide use. Most communications through APIs involve XML or JSON payloads.

Web Services Description Language (WSDL) is the standard way of publishing a SOAP API and there are a range of Web Service specifications.

See also EMML by the Open Mashup Alliance for enterprise mashups.

Criticism

The criticism exists that "Web 2.0" does not represent a new version of the World Wide Web at all, but merely continues to use so-called "Web 1.0" technologies and concepts. Techniques such as AJAX do not replace underlying protocols like HTTP, but add an additional layer of abstraction on top of them. Many of the ideas of Web 2.0 had already been featured in implementations on networked systems well before the term "Web 2.0" emerged. Amazon.com, for instance, has allowed users to write reviews and consumer guides since its launch in

1995, in a form of self-publishing. Amazon also opened its API to outside developers in 2002.[31] Previous developments also came from research in computer-supported collaborative learning and computer-supported cooperative work and from established products like Lotus Notes and Lotus Domino.

In a podcast interview[4], Tim Berners-Lee described the term "Web 2.0" as a "piece of jargon":

> "Nobody really knows what it means...If Web 2.0 for you is blogs and wikis, then that is people to people. But that was what the Web was supposed to be all along."[4]

Other criticism has included the term “a second bubble” (referring to the Dot-com bubble of circa 1995–2001), suggesting that too many Web 2.0 companies attempt to develop the same product with a lack of business models. *The Economist* has also written about "Bubble 2.0".[32] Venture capitalist Josh Kopelman noted that Web 2.0 had excited only 53,651 people (the number of subscribers at that time to TechCrunch, a Weblog covering Web 2.0 startups and technology news), too few users to make them an economically viable target for consumer applications.[33] Although Bruce Sterling reports he's a fan of Web 2.0, he thinks it is now dead as a rallying concept.[34]

Critics have cited the language used to describe the hype cycle of Web 2.0[35] as an example of Techno-utopianist rhetoric.[36]

Critics such as Andrew Keen argue that Web 2.0 has created a cult of digital narcissism and amateurism, which undermines the notion of expertise by allowing anybody, anywhere to share (and place undue value upon) their own opinions about any subject and post any kind of content regardless of their particular talents, knowledgeability, credentials, biases or possible hidden agendas. He states that the core assumption of Web 2.0, that all opinions and user-generated content are equally valuable and relevant is misguided, and is instead "creating an endless digital forest of mediocrity: uninformed political commentary, unseemly home videos, embarrassingly amateurish music, unreadable poems, essays and novels," also stating that Wikipedia is full of "mistakes, half truths and misunderstandings".[37]

Trademark

In November 2004, CMP Media applied to the USPTO for a service mark on the use of the term "WEB 2.0" for live events.[38] On the basis of this application, CMP Media sent a cease-and-desist demand to the Irish non-profit organization IT@Cork on May 24, 2006,[39] but retracted it two days later.[40] The "WEB 2.0" service mark registration passed final PTO Examining Attorney review on May 10, 2006, and was registered on June 27, 2006.[38] The European Union application (application number 004972212, which would confer unambiguous status in Ireland) remains currently pending after its filing on March 23, 2006.

See also

- Buzzword
- Business 2.0
- Cloud computing
- Collective intelligence
- Consumer-generated media
- CreateDebate
- Enterprise 2.0
- Enterprise bookmarking
- Government 2.0

- Learning 2.0
- Library 2.0
- Mashups
- Open Mashup Alliance
- Medicine 2.0
- New Media
- Office suite
- Open source governance
- Radical Trust
- Social commerce
- Social media
- Social networks
- Social shopping
- User-generated content
- Web 1.0
- Web 2.0 for development (web2fordev)
- You (Time Person of the Year)

References

1. ^ "Core Characteristics of Web 2.0 Services (http://www.techpluto.com/web-20-services/)". http://www.techpluto.com/web-20-services/.
2. ^ *a b c* Paul Graham (November 2005). "Web 2.0 (http://www.paulgraham.com/web20.html)". http://www.paulgraham.com/web20.html. Retrieved 2006-08-02. ""I first heard the phrase 'Web 2.0' in the name of the Web 2.0 conference in 2004.""
3. ^ *a b c d e f* Tim O'Reilly (2005-09-30). "What Is Web 2.0 (http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html)". O'Reilly Network. http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html. Retrieved 2006-08-06.
4. ^ *a b c* "developerWorks Interviews: Tim Berners-Lee (http://www.ibm.com/developerworks/podcast/dwi/cm-int082206txt.html)". 2006-07-28. http://www.ibm.com/developerworks/podcast/dwi/cm-int082206txt.html. Retrieved 2007-02-07.
5. ^ DiNucci, D. (1999). "Fragmented Future (http://www.cdinucci.com/Darcy2/articles/Print/Printarticle7.html)". *Print* **53** (4): 32. http://www.cdinucci.com/Darcy2/articles/Print/Printarticle7.html.
6. ^ Idehen, Kingsley. 2003. RSS: INJAN (It's not just about news). Blog. Blog Data Space. August 21. http://www.openlinksw.com/dataspace/kidehen@openlinksw.com/weblog/kidehen@openlinksw.com%27s%20BLOG%20%5B127%5D/241.
7. ^ Idehen, Kingsley. 2003. Jeff Bezos Comments about Web Services. Blog. Blog Data Space. September 25. http://www.openlinksw.com/blog/~kidehen/index.vspx?id=373.
8. ^ Knorr, Eric. 2003. The year of Web services. CIO, December 15.
9. ^ ibid.
10. ^ O'Reilly, Tim, and John Battelle. 2004. Opening Welcome: State of the Internet Industry. In . San Francisco, CA, October 5.
11. ^ Tim O'Reilly (2006-12-10). "Web 2.0 Compact Definition: Trying Again (http://radar.oreilly.com/archives/2006/12/web_20_compact.html)". http://radar.oreilly.com/archives/2006/12/web_20_compact.html. Retrieved 2007-01-20.
12. ^ Grossman, Lev. 2006. Person of the Year: You. December 25. http://www.time.com/time/covers/0,16641,20061225,00.html.
13. ^ *a b* Dion Hinchcliffe (2006-04-02). "The State of Web 2.0 (http://web2.wsj2.com/the_state_of_web_20.htm)". Web Services Journal. http://web2.wsj2.com/the_state_of_web_20.htm. Retrieved 2006-08-06.
14. ^ Bart Decrem (2006-06-13). "Introducing Flock Beta 1 (http://www.flock.com/node/4500)". Flock official blog. http://www.flock.com/node/4500. Retrieved 2007-01-13.
15. ^ Gerald Marwell and Ruth E. Ames: "Experiments on the Provision of Public Goods. I. Resources, Interest, Group Size, and the Free-Rider Problem". *The American Journal of Sociology*, Vol. 84, No. 6 (May, 1979), pp. 1335-1360
16. ^ Best, D., 2006. Web 2.0 Next Big Thing or Next Big Internet Bubble? Lecture Web Information Systems. Techni sche Universiteit Eindhoven.

17. ^ Greenmeier, Larry and Gaudin, Sharon. "Amid The Rush To Web 2.0, Some Words Of Warning -- Web 2.0 -- InformationWeek (http://www.informationweek.com/news/management/showArticle.jhtml;jsessionid=EWRPGLVJ53OW2QSNDLPCKHSCJUNN2JVN?articleID=199702353&_requestid=494050)". www.informationweek.com. http://www.informationweek.com/news/management/showArticle.jhtml;jsessionid=EWRPGLVJ53OW2QSNDLPCKHSCJUNN2JVN?articleID=199702353&_requestid=494050. Retrieved 2008-04-04.
18. ^ O'Reilly, T., 2005. What is Web 2.0. Design Patterns and Business Models for the Next Generation of Software, 30, p.2005
19. ^ McAfee, A. (2006). Enterprise 2.0: The Dawn of Emergent Collaboration. MIT Sloan Management review. Vol. 47, No. 3, p. 21-28.
20. ^ Schick, S., 2005. I second that emotion. IT Business.ca (Canada).
21. ^ Miller, P., 2008. Library 2.0: The Challenge of Disruptive Innovation. Available at: http://www.google.com/url?sa=t&source=web&ct=res&cd=1&url=http%3A%2F%2Fwww.talis.com%2Fresources%2Fdocuments%2F447_Library_2_prf1.pdf&ei=qTUWSsWsH5SJtgeAlNTiDA&usg=AFQjCNER9YGaOBSyJFzz4lQ_GszwcQVqlw-Q
22. ^ Singer, Jonathan B. (2009). *The Role and Regulations for Technology in Social Work Practice and E-Therapy: Social Work 2.0. In A. R. Roberts (Ed). (http://www.us.oup.com/us/catalog/general/subject/SocialWork/~~/dmlldz11c2EmY2k9OTc4MDE5NTM2OTM3Mw==)*. New York, U.S.A.: Oxford University Press. ISBN 978-0195369373. http://www.us.oup.com/us/catalog/general/subject/SocialWork/~~/dmlldz11c2EmY2k9OTc4MDE5NTM2OTM3Mw==.
23. ^ Breakenridge, D., 2008. PR 2.0: New Media, New Tools, New Audiences 1st ed., FT Press.
24. ^ Eggers, William D. (2005). *Government 2.0: Using Technology to Improve Education, Cut Red Tape, Reduce Gridlock, and Enhance Democracy (http://www.manhattan-institute.org/government2.0/)*. Lanham MD, U.S.A.: Rowman & Littlefield Publishers, Inc.. ISBN 978-0742541757. http://www.manhattan-institute.org/government2.0/.
25. ^ Miller 10-11
26. ^ http://news.bbc.co.uk/1/hi/world/americas/8092549.stm
27. ^ "Google buys Web word-processing technology (http://www.news.com/2100-1032_3-6048136.html)". www.news.com. http://www.news.com/2100-1032_3-6048136.html. Retrieved 2007-12-12.
28. ^ "Can eyeOS Succeed Where Desktop.com Failed? (http://www.techcrunch.com/2006/11/27/eyeos-open-source-webos-for-the-masses/)". www.techcrunch.com. http://www.techcrunch.com/2006/11/27/eyeos-open-source-webos-for-the-masses/. Retrieved 2007-12-12.
29. ^ "Tech Beat Hey YouOS! - BusinessWeek (http://www.businessweek.com/the_thread/techbeat/archives/2006/03/hey_youos.html)". www.businessweek.com. http://www.businessweek.com/the_thread/techbeat/archives/2006/03/hey_youos.html. Retrieved 2007-12-12.
30. ^ "PC World — WebEx Snaps Up Intranets.com (http://www.pcworld.com/article/id,122068-page,1/article.html)". www.pcworld.com. http://www.pcworld.com/article/id,122068-page,1/article.html. Retrieved 2007-12-12.
31. ^ Tim O'Reilly (2002-06-18). "Amazon Web Services API (http://www.oreillynet.com/pub/wlg/1707?wlg=yes)". *O'Reilly Network*. http://www.oreillynet.com/pub/wlg/1707?wlg=yes. Retrieved 2006-05-27.
32. ^ "Bubble 2.0 (http://www.economist.com/business/displaystory.cfm?story_id=E1_QQNVDDS)". *The Economist*. 2005-12-22. http://www.economist.com/business/displaystory.cfm?story_id=E1_QQNVDDS. Retrieved 2006-12-20.
33. ^ Josh Kopelman (2006-05-11). "53,651 (http://redeye.firstround.com/2006/05/53651.html)". *Redeye VC*. http://redeye.firstround.com/2006/05/53651.html. Retrieved 2006-12-21.
34. ^ ""Bruce Sterling presenta il web 2.0" (http://www.lastampa.it/multimedia/multimedia.asp?p=1&IDmsezione=29&IDalbum=8558&tipo=VIDEO#mpos)". *"LASTAMPA.it"*. http://www.lastampa.it/multimedia/multimedia.asp?p=1&IDmsezione=29&IDalbum=8558&tipo=VIDEO#mpos.
35. ^ ""Gartner 2006 Emerging Technologies Hype Cycle (http://www.gartner.com/it/page.jsp?id=495475)". http://www.gartner.com/it/page.jsp?id=495475.
36. ^ ""Critical Perspectives on Web 2.0," Special issue of (http://www.uic.edu/htbin/cgiwrap/bin/ojs/index.php/fm/issue/view/263/showToc)First Monday, 13(3), 2008.". http://www.uic.edu/htbin/cgiwrap/bin/ojs/index.php/fm/issue/view/263/showToc.
37. ^ ""Thinking is so over (http://technology.timesonline.co.uk/tol/news/tech_and_web/personal_tech/article1874668.ece)". http://technology.timesonline.co.uk/tol/news/tech_and_web/personal_tech/article1874668.ece.
38. ^ *a b* USPTO serial number 78322306 (http://tarr.uspto.gov/servlet/tarr?regser=serial&entry=78322306)
39. ^ "O'Reilly and CMP Exercise Trademark on 'Web 2.0' (http://yro.slashdot.org/article.pl?sid=06/05/26/1238245)". *Slashdot*. 2006-05-26. http://yro.slashdot.org/article.pl?sid=06/05/26/1238245. Retrieved 2006-05-27.

40. ^ Nathan Torkington (2006-05-26). "O'Reilly's coverage of Web 2.0 as a service mark (http://radar.oreilly.com/archives/2006/05/more_on_our_web_20_service_mar.html)". *O'Reilly Radar*. http://radar.oreilly.com/archives/2006/05/more_on_our_web_20_service_mar.html. Retrieved 2006-06-01.

External links

- Deloitte & Touche LLP - Canada (2008 study) - Change your world or the world will change you: The future of collaborative government and Web 2.0 (http://www.deloitte.com/dtt/article/0,1002,cid=199524,00.html)
- McKinsey & Company - Global Survey - How businesses are using Web 2.0 (http://www.mckinseyquarterly.com/Information_Technology/Applications/How_businesses_are_using_\ gp=1), June 2008
- "Critical Perspectives on Web 2.0" (http://www.uic.edu/htbin/cgiwrap/bin/ojs/index.php/fm/issue/view/263/showToc), Special issue of *First Monday*, 13(3), 2008.
- MacManus, Richard. Porter, Joshua. "Web 2.0 for Designers" (http://www.digital-web.com/articles/web_2_for_designers/). *Digital Web Magazine*, May 4, 2005.
- Graham Vickery, Sacha Wunsch-Vincent: "Participative Web and User-Created Content: Web 2.0, Wikis and Social Networking" (http://www.oecd.org/document/40/0,3343,en_2649_201185_39428648_1_1_1_1,00.html); OECD, 2007
- Harjit Sandhu, CodersAdvocate: "Web 2.0 simplified, where did it all start" (http://www.codersadvocate.com/blogs/44/Web_2.0); 2009

Retrieved from "http://en.wikipedia.org/wiki/Web_2.0"
Categories: Branding | Buzzwords | Neologisms | Cloud applications | Web services | Web 2.0 | Internet memes | Social information processing

Internet2

From Wikipedia, the free encyclopedia

Internet2[1] is an advanced networking consortium led by the research and education community. The not-for-profit partnership includes leaders from research, academia, industry and government. In 2009, Internet2 member rolls included over 30 research and education regional networks known as "connectors" which act as local POPs,[2], with those connectors providing connectivity to over 200 higher education institutions,[3], over 50 affiliate members[4] and over 40 members from industry.[5]

Internet2 operates the Internet2 Network,[6] a next-generation Internet Protocol and optical network that delivers production network services to meet the high-performance demands of research and education, and provides a secure network testing and research environment. In late 2007, Internet2 began operating its newest dynamic circuit network, the Internet2 DCN, an advanced technology that allows user-based allocation of high-capacity data circuits over the fiber-optic network.

The Internet2 Network, through its regional network and connector members, connects over 60,000 U.S. educational, research, government and "community anchor" institutions, from primary and secondary schools to community colleges and universities, public libraries and museums to healthcare organizations.[7]

The Internet2 community is actively engaged in developing and deploying emerging network technologies beyond the scope of single institutions and critical to the future of the Internet. These technologies include large-scale network performance measurement and management tools,[8] simple and secure identity and access management tools[9] and advanced capabilities such as the on-demand creation and scheduling of high-bandwidth, high-performance circuits.[10]

Internet2 is member led and member focused, with an open governance structure and process.[11] Members serve on several advisory councils,[12] collaborate in a variety of working groups and special interest groups[13] gather at spring and fall member meetings,[14] and are encouraged to participate in the strategic planning process.[15]

Contents

History

As the Internet gained in public recognition and popularity, universities were among the first institutions to outgrow the Internet's bandwidth limitations because of the data transfer requirements that many academic researchers need to collaborate with their colleagues. Some universities realized the need for a network that would better support high-performance applications like data mining, medical imaging and particle physics. This resulted in the creation of the very-high-performance Backbone Network Service, or vBNS, developed in 1995 by the National Science Foundation (NSF) and MCI, specifically to meet the needs of the supercomputers at educational institutions. The concept of the "next-generation Internet" was born. After the expiration of the NSF agreement, vBNS largely transitioned to providing service to the government. As a result, the research and education community founded Internet2 to serve its unique networking needs.

The Internet2 Project was originally established by 34 university researchers in 1996 under the auspices of EDUCAUSE and was formally organized as the not-for-profit University Corporation for Advanced Internet Development (UCAID) in 1997, later changing its name to Internet2. Internet2 is a registered trademark.[16] The Internet2 consortium administrative headquarters is located in Ann Arbor, Michigan, with offices in Washington, D.C.[17]

The Internet2 community, in partnership with Qwest, built the first Internet2 Network, called Abilene, in 1998 and was a prime investor in the National LambdaRail (NLR) project.[18]

During 2004–2006, Internet2 and NLR held extensive discussions regarding a possible merger. Those talks paused in spring, 2006, resumed in March, 2007, but eventually ceased in the fall of 2007, due to unresolved differences.[19] [20]

In 2006, Internet2 announced a partnership with Level 3 Communications to launch a brand new nationwide network, boosting its capacity from 10Gbps to 100Gbps.[21] In October, 2007, Internet2 officially retired Abilene and now refers to its new, higher capacity network as the Internet2 Network.

Objectives

Internet2 provides the U.S. research and education community with a network that satisfies their bandwidth-intensive requirements. The network itself is a dynamic, robust and cost-effective hybrid optical and packet network. It furnishes a 100Gb/s network backbone to more than 210 U.S. educational institutions, 70 corporations and 45 non-profit and government agencies.

The Internet2 consortium's objectives are:

- Developing and maintaining a leading-edge network.
- Fully exploiting the capabilities of broadband connections through the use of new-generation applications.
- Transferring new network services and applications to all levels of educational use, and eventually the broader Internet community.

The uses of the network span from collaborative applications, distributed research experiments, grid-based data analysis and social networking. Some of these applications are in varying levels of commercialization, such as IPv6, open-source middleware for secure network access, layer 2 VPNs and dynamic circuit networks.

Achievements

These technologies and their organizational counterparts were not only created to make a faster alternative to

the internet. Many fields have been able to use the Abilene network to foster creativity, research, and development in a way that was not previously possible. Users of poor quality libraries can now download not only text but sound recordings, animations, videos, and other resources, which would be otherwise unavailable. Another application is the robust video conferencing now available to Internet2 participants. Neurosurgeons can now video conference with other experts in the field during an operation in a high resolution format with no apparent time lag.

Application Awards

The *Internet2 Driving Exemplary Applications* (IDEA) award (not to be confused with IDEA awards) was first announced by Internet2 in 2006 as a way of recognizing those who create and use advanced network applications at their best.[22] The judging is conducted by many universities and based upon the following criteria:

- Magnitude of the positive impact of the application for its (current) users
- Technical merit of the application.
- Breadth of impact, as indicated by current user base and likelihood of broader adoption by its full natural community of potential users

Winners are announced each year at the Spring member meeting: 2006,[23] 2007,[24] 2008.[25]

See also

- CANARIE (Canadian research network)
- CLARA (Cooperación Latino Americana de Redes Avanzandas)
- DANTE (Not-for-profit organization managing the pan-European research network)
- DREN (U.S. Department of Defense research and engineering network)
- GEANT (The pan-European research network)
- Kennisnet (Dutch public Internet organization)
- JANET (British academic network)
- Renater (French research network)
- SURFnet (Dutch research network)
- TERENA (European research and education Networks)

References

1. ^ "About Internet2" (http://internet2.edu/about) , Retrieved on 2009-06-26
2. ^ "Internet2 R&E Members List" (http://www.internet2.edu/renm/member.cfm) , Retrieved on 2009-06-26
3. ^ "Internet2 University Members List" (http://members.internet2.edu/university/universities.cfm) , Retrieved on 2009-06-26
4. ^ "Internet2 Affiliate Members List" (http://members.internet2.edu/affiliate/affiliates.cfm) , Retrieved on 2009-06-26
5. ^ "Internet2 Corporate Members" (http://members.internet2.edu/corporate/corporate.cfm) , Retrieved on 2009-06-26
6. ^ "Internet2 Network" (http://www.internet2.edu/network/) , Retrieved on 2009-06-26
7. ^ "Internet2 K20 Connectivity Data" (http://k20.internet2.edu/connectivity) , Retrieved on 2009-06-26
8. ^ "Internet2 Performance Initiative" (http://www.internet2.edu/performance/) , Retrieved on 2009-06-26
9. ^ "Internet2 Security Directory and Related Links" (http://security.internet2.edu/) , Retrieved on 2009-06-26
10. ^ "Internet2 Dynamic Circuit Network" (http://www.internet2.edu/network/dc/) , Retrieved on 2009-06-26

11. ^ "Internet2 Governance" (http://www.internet2.edu/governance/) , Retrieved on 2009-06-26
12. ^ "Internet2 Advisory Councils" (http://www.internet2.edu/governance/advisorycouncils.html) , Retrieved on 2009-06-26
13. ^ "Internet2 Working Groups, SIGs, and Advisory Groups" (http://www.internet2.edu/wg/wg-list.html) , Retrieved on 2009-06-26
14. ^ "Internet2 Member Meetings" (http://www.internet2.edu/meetings/) , Retrieved on 2009-06-26
15. ^ "Internet2 Strategic Planning" (http://www.internet2.edu/strategicplanning/) , Retrieved on 2009-06-26
16. ^ "Internet2 - Terms of Use" (http://www.internet2.edu/termsofuse.html) , Internet2.
17. ^ "Internet2 Contact Information" (http://www.internet2.edu/about/contact/) , Internet2.
18. ^ "Optical networking: The next generation - CNET News.com, By Marguerite Reardon (Staff Writer), Published October 11, 2004 4:00 AM PDT (http://news.com.com/Optical%20networking%20The%20next%20generation/2100-1033-5403589.html?part=dht&tag=ntop&tag=nl.e703)
19. ^ "Internet2, National Lambda Rail, In Merger Talks (July 18, 2005)" (http://www.osc.edu/press/releases/2005/lambda.shtml)
20. ^ Internate archive of now defunct www.internet2-nlr.org site (http://web.archive.org/web/20080327142035/http://www.internet2-nlr.org/)
21. ^ "Speedy Internet2 gets 10x boost - USAToday.com, By Anick Jesdanun (AP Internet Writer), Published October 11, 2007 8:16 AM" (http://www.usatoday.com/tech/news/techinnovations/2007-10-11-faster-internet2_N.htm) , Retrieved on 2009-06-26
22. ^ Internet2 IDEA Awards (http://www.internet2.edu/idea/)
23. ^ IDEA Award Winners 2006 (http://www.internet2.edu/idea/2006/)
24. ^ IDEA Award Winners 2007 (http://www.internet2.edu/idea/2007/index.html)
25. ^ IDEA Award Winners 2008 (http://www.internet2.edu/idea/2008/)

Selected bibliography

- Barnes, Christopher, and Terresa E. Jackson . *INTERNET2: The Backbone of the Future*. Brooks Air Force Base, Tex.: Air Force Research Laboratory, 2002. http://handle.dtic.mil/100.2/ADA406627 (9 February 2007).
- Internet2. *Internet2* - Home. http://www.internet2.edu/ (9 February 2007).
- Matlis, Jan. "Internet2." Computerworld, 28 August 2006, 30.
- Moschovitis, Christos, Hilary Poole, Tami Schuyler, and Theresa M. Senft. *History of the Internet: A Chronology, 1843 to the Present*. Santa Barbara, Cal.: ABC-CLIO, 1999.
- Van Houweling, Douglas and Ted Hanss, "Internet2: The Promise of Truly Advanced Broadband," in The Broadband Explosion, R. Austin and S. Bradley, Editors, Harvard Business School Press, 2005.

External links

- Official website (http://www.internet2.edu/)
  - Internet2 member institution list (http://www.internet2.edu/resources/Internet2MembersList.PDF)
- Internet2 FAQs, from Qwest (http://www.qwest.com/about/qwest/internet2/faqs.html)
- Internetnews.com: Scientists Set Internet2 Speed Record (http://www.internetnews.com/infra/article.php/3403161) 2004-09-02
- Internet2 Talk:The Development of the Internet, Implications for our Future (http://beryl.cse.ucsc.edu/video/Distinguished_Lectures/Houweling2_4.wmv) , by Douglas E Van Houweling PhD, President and CEO of Internet2, at the University of California Santa Cruz, Santa Cruz, California USA, 2009-02-04

Categories: Science and technology in the United States | Internet history | Academic computer network organizations | Consortia

Data Mining: What is Data Mining?

Overview

Generally, data mining (sometimes called data or knowledge discovery) is the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cuts costs, or both. Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases.

Continuous Innovation

Although data mining is a relatively new term, the technology is not. Companies have used powerful computers to sift through volumes of supermarket scanner data and analyze market research reports for years. However, continuous innovations in computer processing power, disk storage, and statistical software are dramatically increasing the accuracy of analysis while driving down the cost.

Example

For example, one Midwest grocery chain used the data mining capacity of Oracle software to analyze local buying patterns. They discovered that when men bought diapers on Thursdays and Saturdays, they also tended to buy beer. Further analysis showed that these shoppers typically did their weekly grocery shopping on Saturdays. On Thursdays, however, they only bought a few items. The retailer concluded that they purchased the beer to have it available for the upcoming weekend. The grocery chain could use this newly discovered information in various ways to increase revenue. For example, they could move the beer display closer to the diaper display. And, they could make sure beer and diapers were sold at full price on Thursdays.

Data, Information, and Knowledge

Data

Data are any facts, numbers, or text that can be processed by a computer. Today, organizations are accumulating vast and growing amounts of data in different formats and different databases. This includes:

- operational or transactional data such as, sales, cost, inventory, payroll, and accounting
- nonoperational data, such as industry sales, forecast data, and macro economic data
- meta data - data about the data itself, such as logical database design or data dictionary definitions

Information

The patterns, associations, or relationships among all this *data* can provide *information*. For example, analysis of retail point of sale transaction data can yield information on which products are selling and when.

Knowledge

Information can be converted into *knowledge* about historical patterns and future trends. For example, summary information on retail supermarket sales can be analyzed in light of promotional efforts to provide knowledge of

consumer buying behavior. Thus, a manufacturer or retailer could determine which items are most susceptible to promotional efforts.

Data Warehouses

Dramatic advances in data capture, processing power, data transmission, and storage capabilities are enabling organizations to integrate their various databases into *data warehouses*. Data warehousing is defined as a process of centralized data management and retrieval. Data warehousing, like data mining, is a relatively new term although the concept itself has been around for years. Data warehousing represents an ideal vision of maintaining a central repository of all organizational data. Centralization of data is needed to maximize user access and analysis. Dramatic technological advances are making this vision a reality for many companies. And, equally dramatic advances in data analysis software are allowing users to access this data freely. The data analysis software is what supports data mining.

What can data mining do?

Data mining is primarily used today by companies with a strong consumer focus - retail, financial, communication, and marketing organizations. It enables these companies to determine relationships among "internal" factors such as price, product positioning, or staff skills, and "external" factors such as economic indicators, competition, and customer demographics. And, it enables them to determine the impact on sales, customer satisfaction, and corporate profits. Finally, it enables them to "drill down" into summary information to view detail transactional data.

With data mining, a retailer could use point-of-sale records of customer purchases to send targeted promotions based on an individual's purchase history. By mining demographic data from comment or warranty cards, the retailer could develop products and promotions to appeal to specific customer segments.

For example, Blockbuster Entertainment mines its video rental history database to recommend rentals to individual customers. American Express can suggest products to its cardholders based on analysis of their monthly expenditures.

WalMart is pioneering massive data mining to transform its supplier relationships. WalMart captures point-of-sale transactions from over 2,900 stores in 6 countries and continuously transmits this data to its massive 7.5 terabyte Teradata data warehouse. WalMart allows more than 3,500 suppliers, to access data on their products and perform data analyses. These suppliers use this data to identify customer buying patterns at the store display level. They use this information to manage local store inventory and identify new merchandising opportunities. In 1995, WalMart computers processed over 1 million complex data queries.

The National Basketball Association (NBA) is exploring a data mining application that can be used in conjunction with image recordings of basketball games. The Advanced Scout software analyzes the movements of players to help coaches orchestrate plays and strategies. For example, an analysis of the play-by-play sheet of the game played between the New York Knicks and the Cleveland Cavaliers on January 6, 1995 reveals that when Mark Price played the Guard position, John Williams attempted four jump shots and made each one! Advanced Scout not only finds this pattern, but explains that it is interesting because it differs considerably from the average shooting percentage of 49.30% for the Cavaliers during that game.

By using the NBA universal clock, a coach can automatically bring up the video clips showing each of the jump shots attempted by Williams with Price on the floor, without needing to comb through hours of video footage. Those clips show a very successful pick-and-roll play in which Price draws the Knick's defense and then finds Williams for an open jump shot.

How does data mining work?

While large-scale information technology has been evolving separate transaction and analytical systems, data mining provides the link between the two. Data mining software analyzes relationships and patterns in stored transaction data based on open-ended user queries. Several types of analytical software are available: statistical, machine learning, and neural networks. Generally, any of four types of relationships are sought:

- **Classes**: Stored data is used to locate data in predetermined groups. For example, a restaurant chain could mine customer purchase data to determine when customers visit and what they typically order. This information could be used to increase traffic by having daily specials.
- **Clusters**: Data items are grouped according to logical relationships or consumer preferences. For example, data can be mined to identify market segments or consumer affinities.
- **Associations**: Data can be mined to identify associations. The beer-diaper example is an example of associative mining.
- **Sequential patterns**: Data is mined to anticipate behavior patterns and trends. For example, an outdoor equipment retailer could predict the likelihood of a backpack being purchased based on a consumer's purchase of sleeping bags and hiking shoes.

Data mining consists of five major elements:

- Extract, transform, and load transaction data onto the data warehouse system.
- Store and manage the data in a multidimensional database system.
- Provide data access to business analysts and information technology professionals.
- Analyze the data by application software.
- Present the data in a useful format, such as a graph or table.

Different levels of analysis are available:

- **Artificial neural networks**: Non-linear predictive models that learn through training and resemble biological neural networks in structure.
- **Genetic algorithms**: Optimization techniques that use processes such as genetic combination, mutation, and natural selection in a design based on the concepts of natural evolution.
- **Decision trees**: Tree-shaped structures that represent sets of decisions. These decisions generate rules for the classification of a dataset. Specific decision tree methods include Classification and Regression Trees (CART) and Chi Square Automatic Interaction Detection (CHAID) . CART and CHAID are decision tree techniques used for classification of a dataset. They provide a set of rules that you can apply to a new (unclassified) dataset to predict which records will have a given outcome. CART segments a dataset by creating 2-way splits while CHAID segments using chi square tests to create multi-way splits. CART typically requires less data preparation than CHAID.
- **Nearest neighbor method**: A technique that classifies each record in a dataset based on a combination of the classes of the *k* record(s) most similar to it in a historical dataset (where *k* 1). Sometimes called the *k*-

nearest neighbor technique.

- **Rule induction**: The extraction of useful if-then rules from data based on statistical significance.

- **Data visualization**: The visual interpretation of complex relationships in multidimensional data. Graphics tools are used to illustrate data relationships.

What technological infrastructure is required?

Today, data mining applications are available on all size systems for mainframe, client/server, and PC platforms. System prices range from several thousand dollars for the smallest applications up to $1 million a terabyte for the largest. Enterprise-wide applications generally range in size from 10 gigabytes to over 11 terabytes. NCR has the capacity to deliver applications exceeding 100 terabytes. There are two critical technological drivers:

- **Size of the database**: the more data being processed and maintained, the more powerful the system required.

- **Query complexity**: the more complex the queries and the greater the number of queries being processed, the more powerful the system required.

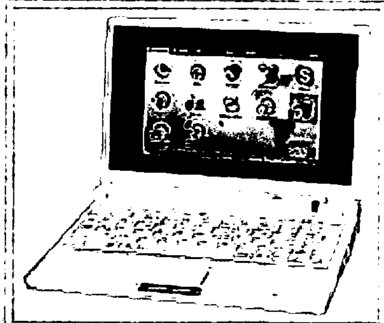
Relational database storage and management technology is adequate for many data mining applications less than 50 gigabytes. However, this infrastructure needs to be significantly enhanced to support larger applications. Some vendors have added extensive indexing capabilities to improve query performance. Others use new hardware architectures such as Massively Parallel Processors (MPP) to achieve order-of-magnitude improvements in query time. For example, MPP systems from NCR link hundreds of high-speed Pentium processors to achieve performance levels exceeding those of the largest supercomputers.

[Overview] [What is Data Mining?] [Issues] [More Information]

Netbook

From Wikipedia, the free encyclopedia

A **netbook** is a laptop computer designed for wireless communication and access to the Internet.[1]



An ASUS Eee PC netbook.

Primarily designed for web browsing and e-mailing, netbooks are used on the Internet for remote access to web-based applications[2] and are targeted increasingly at cloud computing users who require a less powerful client computer.[3] Netbooks typically run either Windows XP or Linux operating systems[2] rather than more resource-intensive operating systems like Windows Vista.[4][5] However Windows 7 has seen some features that have been shown to run well on the newer generation of netbooks being released currently. The devices range in size from below 5 inches[6] to over 13,[7] typically weigh 2 to 3 pounds (~1 kg) and are often significantly cheaper than general purpose laptops at US$ 400,[2] with some even given away for free in Japan if the purchaser signs up for a plan.[8]

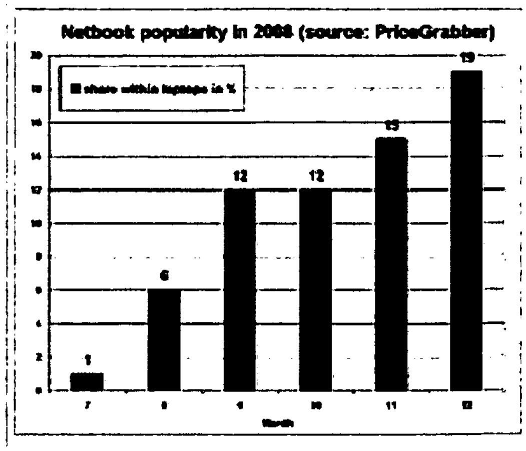
The word netbook is a portmanteau of the words *Internet* and *notebook*.

Contents

History

The origins of the netbook can be traced to the Network Computer (NC) concept of the mid-1990s. More recently, Psion's now-discontinued netBook line, the One Laptop Per Child (OLPC) project and the Palm Foleo were all small,

portable, network-enabled computers.[9][10][11] The generic use of the term "netbook", however, began in 2007 when Asus unveiled the ASUS Eee PC. Originally designed for emerging markets, the 8.9 × 6.5 in (23 x 17 cm) device weighed about two pounds and featured a 7-inch display, a keyboard approximately 85% the size of a normal keyboard, and a custom version of Linux with a simplified user interface geared towards netbook use.[10] Following the Eee PC, Everex launched its Linux-based CloudBook, MSI released the Wind, Dell and HP both released a "Mini" series (the Inspiron Mini and HP Mini), and others soon followed suit. Windows XP models were also introduced.[10]



Netbook market popularity within laptops in second half of 2008 based on the number of product clicks in the Laptop Subcategory per month by PriceGrabber[2]

By late 2008, netbooks had begun to take market share away from laptops.[12] In contrast to earlier, largely failed attempts to establish mini computers as a new class of mainstream personal computing devices built around comparatively expensive platforms requiring proprietary software applications or imposing severe usability limitations, the recent success of netbooks can also be attributed to the fact that PC technology has now matured enough to allow truly cost optimized implementations with enough performance to suit the needs of a majority of PC users. This is illustrated by the fact that typical system performance of a netbook is on the level of a mainstream PC in 2001, at around one quarter of the cost. While this performance level suffices for most of the user needs, it caused an increased interest in resource-efficient applications such as Google's Chrome or Mozilla's Firefox, and forced Microsoft to extend availability of Windows XP in order to secure market share. It is estimated that almost thirty times more netbooks were sold in 2008 (11.4 million, 70% of which were in Europe[13]) than in 2007 (400,000).[14] For 2009, sales are expected to jump to 35 million, rising to an estimated 139 million in 2013.[15] This trend is reinforced by the rise of web-based applications as well as mobile networking and, according to Wired Magazine, netbooks are evolving into "super-portable laptops for professionals".[16]

Microsoft and Intel have tried to "cement" netbooks in the low end of the market to protect mainstream notebook PC sales, because they get lower margins on low-cost models. The companies have limited the specifications of netbooks, but despite this original equipment manufacturers have announced higher-end netbooks models as of March 2009.[17]

Trademarks

In 1996 Psion started applying for trademarks for a line of *netBook* products that was later released in 1999.[18] International trademarks were issued (including U.S. Trademark 75,215,401 (http://tarr.uspto.gov/servlet/tarr?regser=serial&entry=75215401) and Community Trade Mark 000428250 (http://oami.europa.eu/CTMOnline/RequestManager/en_Result?transition=ResultsDetailed&ntmark=&application=CTMOnline&bAdvanced=0&language=en&deno=&source= but the models failed to gain popularity[19] and are now discontinued (except for providing accessories, maintenance and support to existing users).[20] Similar marks were recently rejected by the USPTO citing a "likelihood of confusion" under section 2(d).[21][22][23]

Despite expert analysis that the mark is "probably generic",[24] Psion Teklogix issued cease and desist letters on 23 December, 2008.[25][26][27] This was heavily criticised,[28][29][30] prompting the formation of the "Save the Netbooks (http://www.savethenetbooks.com/)" grassroots campaign which worked to reverse the Google AdWords ban, cancel the trademark and encourage continued generic use of the term.[31] While preparing a

MIPS

Some Ultra-Low Cost netbooks feature a MIPS CPU. [47] The 64-bit Loongson MIPS microprocessor is also used for higher-end applications. [48]

x86

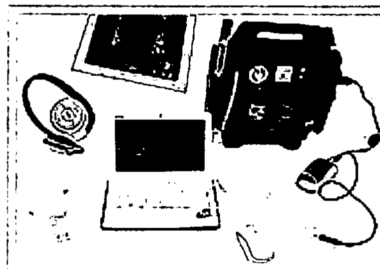
One report at the end of 2008 suggested the typical netbook featured a 3-lb (1.4 kg) weight, a 9-inch (23 cm) screen, wireless Internet connectivity, Linux or Windows XP, an Intel chip, and a cost of less than US$ 400.[49] The x86-compatible VIA Technologies C7 processor is powering netbooks from HP[50] and Samsung.[51] VIA has also designed the Nano, a new x86-64-compatible architecture targeting lower priced, mobile applications like netbooks.

Software

Windows

As of January 2009, over 90% of netbooks are estimated to ship with Windows XP[52], which Microsoft was later estimated to sell ranging from US$15 to US$ 35 per netbook.[53][54] Microsoft has extended the availability of Windows XP for ultra-low cost personal computers from June 2008 until June 2010,[55] possibly to keep netbooks from gaining market share at the expense of desktops and "value" laptops[56] and to avoid increased use of Linux installations on netbooks.[57]

Microsoft is also testing[58] and has demonstrated[59] a 'Starter' edition of Windows 7 for this class of devices.[60][61] Windows CE has also been used in netbook applications, due to its reduced feature design, that keeps with the design philosophy of netbooks. [62]


Netbook with Accessories

Linux

As of January 2009, customised Linux distributions are estimated to ship on less than 10% of netbooks,[52] making it the second most popular operating system after Windows. As Linux systems normally install software from an Internet software repository, they do not need an optical drive to install software. However, early netbooks like the Eee PC failed to use this benefit by disabling access to the full range of available Linux software.

Netbooks have sparked the development of many new Linux distributions, like Ubuntu Netbook Remix and Easy Peasy. See the full list of Netbook Distributions. Recently an Intel-sponsored beta version of Moblin has hit the web.

Google Chrome OS

On July 7, 2009, Google announced via a blog post that it was developing Google Chrome OS, an operating system which combines the Google Chrome web browser with the Linux kernel, to be "targeted at Netbooks." [63]

Android

Google's Android software platform, designed for mobile telephone handsets, has been demonstrated on an

"Petition for Cancellation" of U.S. Trademark 75,215,401 (http://tarr.uspto.gov/servlet/tarr?regser=serial&entry=75215401) they revealed[32] that Dell had submitted one the day before[33] on the basis of abandonment, genericness and fraud.[34] They later revealed Psion's counter-suit against Intel, filed on 27 February 2009.[35]

It was also revealed around the same time that Intel had also sued Psion Teklogix (US & Canada) and Psion (UK) in the Federal Court on similar grounds.[36] In addition to seeking cancellation of the trademark, Intel sought an order enjoining Psion from asserting any trademark rights in the term "netbook", a declarative judgement regarding their use of the term, attorneys' fees, costs and disbursements and "such other and further relief as the Court deems just and proper".[37]

On June 2, 2009, Psion announced that the suit had been settled out of court. Psion's statement said that the company was withdrawing all of its trademark registrations for the term "Netbook" and that Psion agreed to "waive all its rights against third parties in respect of past, current or future use" of the term.[38]

Technology

Hardware

Netbook users typically rely on online applications and services which do not require powerful hardware on the local computer.[39] Some netbooks do not even have conventional hard[40] or optical disc drives. Such netbooks use solid-state storage devices instead, as these require less power and are smaller, lighter and generally more robust and durable. On machines with no optical disk drive, application software is typically downloaded from the web or read from a USB device. An external IDE drive can be used with a USB-to-IDE converter.

A netbook motherboard featuring the Intel Atom processor

All netbooks on the market today support Wi-Fi wireless networking and many can be used on mobile telephone networks with data capability. Mobile data plans are supplied under contract in the same way as mobile telephone [41]. Some also include Ethernet and/or modem ports, for broadband or dial-up Internet access, respectively.

Processor architectures

ARM

ARM is designing and licensing high-performance chips requiring a relatively low power envelope, which would constitute an ideal basis for netbooks, and has touted these as an alternative platform.[42] Despite this, ARM has had very little success in establishing a market for their chips in netbooks, likely because of incompatibilities of their ARM architecture to the established x86 software ecosystem (primarily the dominant Microsoft Windows operating system, Linux is fully compatible). Freescale, a manufacturer of ARM chips, has projected that by 2012 half of all netbooks will run on ARM[43]. Google has announced they will release a specific netbook OS called Google Chrome OS end 2010 that will support both the x86 and the ARM architecture [44], and there has been much speculation as to a version of the upcoming Windows 7 compatible with ARM.[45] In June 2009 nVidia announced a dozen mobile Internet devices running Tegra, some of which will be netbooks.[46]

See also

- Comparison of netbooks
- Nettop, a desktop equivalent of the netbook
- Smartbook
- Subnotebook
- Operating systems
  - FreeBSD / NetBSD
  - Linux
  - MS-Windows XP

References

1. ^ Intel.com: Thoughts on Netbooks (http://blogs.intel.com/technology/2008/03/thoughts_on_netbooks.php)
2. ^ a b c d *Netbook Trends and Solid-State Technology Forecast (https://mr.pricegrabber.com/Netbook_Trends_and_SolidState_Technology_January_2009_CBR.pdf)*. pricegrabber.com. pp. 7. https://mr.pricegrabber.com/Netbook_Trends_and_SolidState_Technology_January_2009_CBR.pdf. Retrieved on 2009-01-28.
3. ^ "Disruptor: The 'netbook' revolution (http://money.cnn.com/2008/10/13/technology/copeland_asus.fortune/index.htm)". Fortune Magazine, Michael Copeland, October 16, 2008. http://money.cnn.com/2008/10/13/technology/copeland_asus.fortune/index.htm.
4. ^ "Cheap PCs Weigh on Microsoft (http://blogs.wsj.com/biztech/2008/12/08/cheap-pcs-weigh-on-microsoft/)". Business Technologies, The Wall Street Journal, December 8, 2008. http://blogs.wsj.com/biztech/2008/12/08/cheap-pcs-weigh-on-microsoft/.
5. ^ "ASK NBB #3: Netbooks of the Future (http://netbookboards.com/ask-nbb/ask-nbb-3-netbooks-of-the-future/)". NetbookBoards, April 24, 2009. http://netbookboards.com/ask-nbb/ask-nbb-3-netbooks-of-the-future/.
6. ^ UMID Netbook Only 4.8□(http://www.elitezoom.com/umid-netbook-only-48-display.html)
7. ^ CES 2009 - MSI Unveils the X320 “MacBook Air Clone” Netbook (http://www.futurelooks.com/ces-2009-msi-unveils-the-x320-macbook-air-clone-netbook/)
8. ^ "Light and Cheap, Netbooks Are Poised to Reshape PC Industry" (http://www.nytimes.com/2009/04/02/technology/02netbooks.html) (*The New York Times* - April 1, 2009: "AT&T announced on Tuesday that customers in Atlanta could get a type of compact PC called a netbook for just US$50 if they signed up for an Internet service plan..." - “The era of a perfect Internet computer for US$ 99 is coming this year,” said Jen-Hsun Huang, the chief executive of Nvidia, a maker of PC graphics chips that is trying to adapt to the new technological order.
9. ^ "Jeff Hawkins and the World's First Netbook (http://www.pcmag.com/article2/0,2817,2335072,00.asp)". PCMagazine.com, Tim Bajarin, November 21, 2008. http://www.pcmag.com/article2/0,2817,2335072,00.asp.
10. ^ a b c "Netbooks vs. Notebooks (http://www.pcmag.com/article2/0,2817,2330274,00.asp)". PCMagazine.com, Tim Bajarin, September 12, 2008. http://www.pcmag.com/article2/0,2817,2330274,00.asp.
11. ^ Psion's NetBook Pro Abandons EPOC in Favor of Windows CE (http://www.technewsworld.com/story/31758.html?wlc=1235434538)
12. ^ Ian Lamont (2008-10-04). ""Netbooks" Move Up In Notebook Rankings (http://hardware.slashdot.org/article.pl?sid=08/10/04/1827200)". *Slashdot*. http://hardware.slashdot.org/article.pl?sid=08/10/04/1827200. Retrieved on 2008-10-23.
13. ^ 70% of Netbook sales are in Europe (http://www.umpcportal.com/2008/11/70-of-netbook-sales-are-in-europe)
14. ^ Analysis: Did Intel underestimate netbook success? (http://www.eetimes.com/news/latest/showArticle.jhtml?articleID=212701251)
15. ^ ABI Research: 35 Million Netbook Shipments Expected in 2009: An Era Begins (http://www.abiresearch.com/press/1355-35+Million+Netbook+Shipments+Expected+in+2009:+An+Era+Begins)
16. ^ Puny, Trendy Netbooks Are Growing Up to Suit Business Users (http://blog.wired.com/gadgets/2009/01/cheap-netbooks.html)
17. ^ http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9130757

ASUS Eee PC and its Linux operating system contains policies for mobile internet devices including the original Asus Eee PC 701.[64] ASUS has allocated engineers to develop an Android-based netbook.[65] Freescale have also announced plans for a low-cost ARM-based netbook design, running Android.[66] In May 2009 a contractor of Dell announced it is porting Adobe Flash Lite to Android for Dell netbooks.[67] Acer announced Android netbooks to be available in Q3/2009.[68]

Mac OS X

Mac OS X has been demonstrated running on various netbooks as a result of the OSx86 project,[69] although this may be in violation of the operating system's End User License Agreement.[70] Apple has complained to sites hosting information on how to install OS X onto non-Apple hardware (including Wired and YouTube) who have reacted and removed content in response.[71] Although Apple has previously taken legal action regarding such installations (see Psystar), a February 2009 federal court ruling may result in the eventual removal of this restriction.[72]

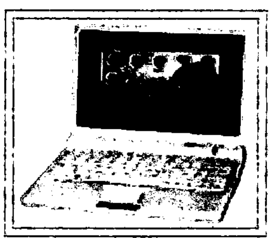
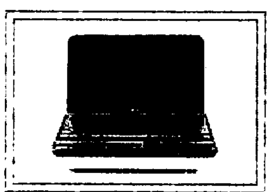
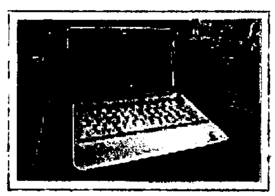
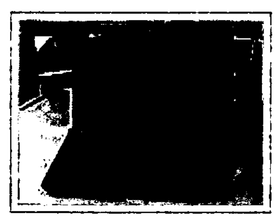
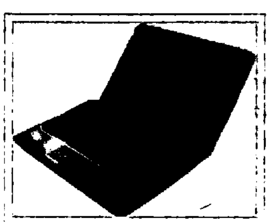
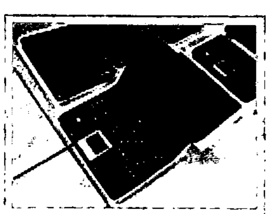
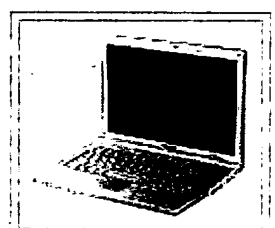
Other

Netbooks have been demonstrated running other operating systems including FreeBSD, OpenBSD, Darwin and Moblin.

Usage

A NPD study found that 60 percent of netbook buyers never take their netbooks out of the house.[73]

Gallery

			
ASUS Eee PC 700	HP 2133 Mini-Note PC	MSI Wind	Acer Aspire One
			
Dell Inspiron Mini 9	Zelybron Micro Nina	Lenovo IdeaPad S10	Samsung NC20

49. ^ "Tech's hope in 2009 - or curse? (http://money.cnn.com/2008/12/24/technology/fortt_netbooks.fortune/index.htm)". Fortune Magazine, December 24, 2008, Jon Fortt. http://money.cnn.com/2008/12/24/technology/fortt_netbooks.fortune/index.htm.
50. ^ Crothers, Brooke (2009-04-09). "Why HP chose Via chip over Intel for Mini-Note (http://news.cnet.com/8301-13924_3-9914927-64.html)". CNET News. http://news.cnet.com/8301-13924_3-9914927-64.html. Retrieved on 2009-04-09.
51. ^ "Samsung's VIA-powered NC20 netbook now available (http://www.electronista.com/articles/09/04/03/samsung.nc20.now.available/)". Electronista. 2009-04-03. http://www.electronista.com/articles/09/04/03/samsung.nc20.now.available/. Retrieved on 2009-04-09.
52. ^ *a b* Think Linux Rules on Netbooks? Think Again (http://www.pcworld.com/businesscenter/article/160567/think_linux_rules_on_netbooks_think_again.html)
53. ^ Microsoft shares hit 11-year low (http://news.zdnet.com/2100-9595_22-272793.html)
54. ^ http://online.wsj.com/article/SB124018108488732939.html
55. ^ Microsoft Announces Extended Availability of Windows XP Home for ULCPCs (http://www.microsoft.com/presspass/features/2008/apr08/04-03xpeos.mspx), April 3, 2008 Press release
56. ^ Microsoft to limit capabilities of cheap laptops (http://www.itworld.com/Comp/2218/microsoft-windows-on-lowcost-pcs-080512/), IT World May 12, 2008
57. ^ Microsoft U-turn to stop Linux dominating ultra low cost PC (http://www.computerworlduk.com/technology/hardware/laptops/news/index.cfm?newsid=9006)
58. ^ Microsoft seeking Win 7 testers for netbooks? (http://blogs.zdnet.com/microsoft/?p=1838)
59. ^ Ars@PDC: Steven Sinofsky on Windows 7 and netbooks (http://arstechnica.com/journals/microsoft.ars/2008/10/28/arspdc-steven-sinofsky-on-windows-7-and-netbooks)
60. ^ Windows 7 to Ship In Six Different Versions (http://www.pcworld.com/article/158861/windows_7_to_ship_in_six_different_versions.html)
61. ^ Confirmed: Windows 7 'netbook edition' (http://crave.cnet.co.uk/laptops/0,39029450,49300490,00.htm)
62. ^ Windows CE takes on Linux in low-end netbooks (http://www.windowsfordevices.com/news/NS6976598029.html)
63. ^ http://googleblog.blogspot.com/2009/07/introducing-google-chrome-os.html
64. ^ Android netbooks on their way, likely by 2010 (http://venturebeat.com/2009/01/01/android-netbooks-on-their-way-likely-by-2010/)
65. ^ Google Android May Run Asus Netbook, Rival Microsoft (Update1) (http://www.bloomberg.com/apps/news?pid=newsarchive&sid=a4E9sAyqigOU)
66. ^ Freescale to use Android, ARM for US$ 100 Netbook (http://blogs.zdnet.com/gadgetreviews/?p=1563)
67. ^ http://www.fiercewireless.com/press-releases/bsquare-port-adobe-flash-lite-new-google-android-netbook-del?utm_medium=nl&utm_source=internal
68. ^ http://www.reuters.com/article/marketsNews/idUSTP18980620090602
69. ^ Mac OS X Netbook Compatibility Chart (Updated) (http://gadgets.boingboing.net/2008/12/17/osx-netbook-compatib.html)
70. ^ It Lives! Gadget Lab's Netbook Running OS X Leopard (http://blog.wired.com/gadgets/2008/09/it-lives-gadget.html)
71. ^ Gadget Lab Video: Running OS X on a Netbook (http://blog.wired.com/gadgets/2008/12/gadget-lab-vide.html)
72. ^ Mac clone maker wins legal round against Apple (http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9127579)
73. ^ http://www.npd.com/press/releases/press_090622b.html

External links

- "The rise of the Netbook" (http://news.cnet.com/Year-in-review-The-rise-of-the-Netbook/2009-1042_3-6248307.html) article at Cnet
- "The State of the Netbook" (http://arstechnica.com/gadgets/news/2009/02/state-of-the-netbook-1.ars) article at Ars Technica
- "The Netbook Effect: How Cheap Little Laptops Hit the Big Time" (http://www.wired.com/gadgets/wireless/magazine/17-03/mf_netbooks?currentPage=all) article at Wired.com
- "Light and Cheap, Netbooks Are Poised to Reshape PC Industry" (http://www.nytimes.com/2009/04/02/technology/02netbooks.html?pagewanted=1&em) article

18. ^ Psion netbook news release (http://www.prnewswire.co.uk/cgi/release?id=16302)
19. ^ Save the Netbooks: fighting a trademark on extinct hardware (http://arstechnica.com/gadgets/news/2009/02/save-the-netbooks-fighting-a-trademark-on-extinct-hardware.ars)
20. ^ Psion Teklogix Discontinued Products (http://www.psionteklogix.com/Public.aspx?s=us&p=EOLProducts)
21. ^ U.S. Trademark 77,527,311 (http://tarr.uspto.gov/servlet/tarr?regser=serial&entry=77527311) for 'G NETBOOK' rejected 31 October 2008
22. ^ U.S. Trademark 77,580,272 (http://tarr.uspto.gov/servlet/tarr?regser=serial&entry=77580272) for MSI's 'WIND NETBOOK'
23. ^ U.S. Trademark 77,590,174 (http://tarr.uspto.gov/servlet/tarr?regser=serial&entry=77590174) for Coby Electronics' 'COBY NETBOOK' rejected 13 January 2009
24. ^ How To Lose A Trademark: "Netbook" Is Probably Generic (http://patenthands.com/ph-blog/how-to-lose-a-trademark-netbook-is-probably-generic/)
25. ^ Netbook enthusiast web sites getting C & D using term "netbook" (http://jkontherun.com/2008/12/23/netbook-enthusiast-web-sites-getting-c-d-using-term-netbook/)
26. ^ http://www.prnewswire.co.uk/cgi/release?id=16302
27. ^ Psion threatens netbook sites over trademarks (http://www.guardian.co.uk/technology/blog/2008/dec/24/psion-netbook-threats)
28. ^ 'Netbook' trademarked already, we're all doomed (http://www.crunchgear.com/2008/12/24/netbook-trademarked-already-were-all-doomed/)
29. ^ Using the Word 'Netbook' Could Get You Sued (http://blog.wired.com/gadgets/2008/12/using-the-word.html)
30. ^ Cease and Desist: the netbook war of words (http://www.itwire.com/content/view/22488/53/)
31. ^ Save the Netbooks: fighting a trademark on extinct hardware (http://arstechnica.com/gadgets/news/2009/02/save-the-netbooks-fighting-a-trademark-on-extinct-hardware.ars)
32. ^ Dell accuses Psion of "fraud" over netbook claims (http://www.pcpro.co.uk/news/247933/dell-accuses-psion-of-fraud-over-netbook-claims.html)
33. ^ Dell fights back against Psion netBook trademark rampage (http://arstechnica.com/gadgets/news/2009/02/dell-fights-back-against-psion-netbook-trademark-rampage.ars)
34. ^ Dell Goes 'Nuclear' Over Netbook Trademark (http://www.crn.com/mobile/214502133)
35. ^ Newsflash: Intel counter-sued by Psion in "netbook" trademark lawsuit; jury trial demanded (http://blog.savethenetbooks.com/2009/03/newsflash-intel-counter-sued-by-psion.html)
36. ^ Intel Wants 'Netbook' Trademark Canceled (http://www.courthousenews.com/2009/02/16/Intel_Wants_Netbook_Trademark_Canceled.htm)
37. ^ Complaint for Injunctive Relief, Declaratory Judgment & Cancellation of Federal Trademark (http://www.courthousenews.com/2009/02/16/IntelvPsion.pdf)
38. ^ Psion, Intel settle 'Netbook' trademark dispute (http://news.cnet.com/8301-1001_3-10253210-92.html)
39. ^ Shoot For the Clouds (http://www.pcworld.com/businesscenter/article/151950/shoot_for_the_clouds.html)
40. ^ What is a Netbook computer? (http://news.cnet.com/8301-13554_3-10063963-33.html)
41. ^ Ganapati, Priya (2008-12-15). "The Next Netbook Trend: Cellphone-Like Contract Deals (http://blog.wired.com/gadgets/2008/12/next-time-you-s.html)". Wired (magazine). http://blog.wired.com/gadgets/2008/12/next-time-you-s.html. Retrieved on 2009-05-20.
42. ^ Fermoso, Jose (2009-03-02). "'Touch Book' Brings Netbook and Tablets Together With Detachable Parts (http://blog.wired.com/gadgets/2009/03/touch-book-comb.html)". http://blog.wired.com/gadgets/2009/03/touch-book-comb.html. Retrieved on 2009-04-09.
43. ^ Tofel, Kevin C. (2009-02-18). jkOnTheRun. http://jkontherun.com/2009/02/18/freescale-says-arm-will-power-half-the-netbook-market/. Retrieved on 2009-04-09.
44. ^ introducing google Chrome OS (http://googleblog.blogspot.com/2009/07/introducing-google-chrome-os.html)
45. ^ Vance, Ashlee (2009-04-06). "Microsoft's Netbook Conundrum (http://bits.blogs.nytimes.com/2009/04/06/microsofts-netbook-conundrum/)". New York Times. http://bits.blogs.nytimes.com/2009/04/06/microsofts-netbook-conundrum/. Retrieved on 2009-04-09.
46. ^ http://news.prnewswire.com/ViewContent.aspx?ACCT=109&STORY=/www/story/06-02-2009/0005036371&EDATE=
47. ^ Hermann, John (2008-12-09). "$170 Alpha 400 MIPS Netbook is as Expensive as it is Desirable (http://gizmodo.com/5105146/170-alpha-400-mips-netbook-is-as-expensive-as-it-is-desirable/)". Gizmodo. http://gizmodo.com/5105146/170-alpha-400-mips-netbook-is-as-expensive-as-it-is-desirable/. Retrieved on 2009-04-09.
48. ^ "The Gdium Liberty 1000 clearly stands out from the pack - Gdium Products (http://www.gdium.com/product/liberty1000)". Gdium. http://www.gdium.com/product/liberty1000. Retrieved on 2009-04-20.

at New York Times
- "Netbook Sales Keep Quanta Competitive" (http://www.businessweek.com/magazine/content/09_22/b4133043584164.htm) article at BusinessWeek
- Netbooks: Small but disruptive | The Economist (http://www.economist.com/displayStory.cfm?story_id=13832588)
- Rational consumer: Small is beautiful | The Economist (http://www.economist.com/displayStory.cfm?story_id=12673233)
- Taiwan Review/High-tech Success Made Simple (http://taiwanreview.nat.gov.tw/ct.asp?xItem=44916&CtNode=119)
- Acer Founder Stan Shih:The Netbook Is the Big Chance for Brand Taiwan (http://english.cw.com.tw/article.do?action=show&id=10865)
- World hardware: Small wonder/THE ECONOMIST INTELLIGENCE UNIT (http://english.cw.com.tw/article.do?action=show&id=10673&offset=0)

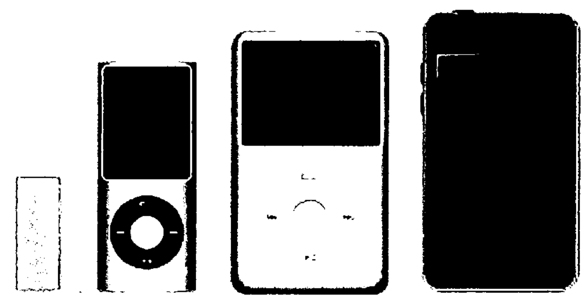
Retrieved from "http://en.wikipedia.org/wiki/Netbook"
Categories: Netbooks | Appropriate technology | Cloud clients
Hidden categories: Wikipedia pages move-protected due to vandalism | All articles with unsourced statements | Articles with unsourced statements from March 2009

---

iPod

From Wikipedia, the free encyclopedia

iPod



The current iPod line. From left to right: iPod Shuffle, iPod Nano, iPod Classic, iPod Touch.

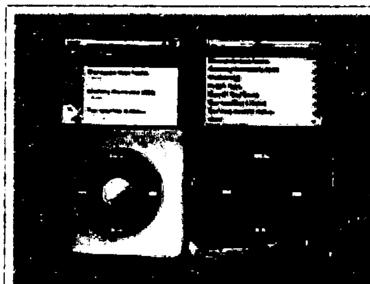
Manufacturer	Apple Inc.
Type	Portable Media Player (PMP)
Units sold	Over 206,000,000 worldwide as of April 2008
CPU	Samsung ARM
Online services	iTunes WiFi Music Store (iPod touch only) App Store(iPod touch only)

iPod is a brand of portable media players designed and marketed by Apple Inc. and launched on October 23, 2001. The product line-up includes the hard drive-based iPod Classic, the touchscreen iPod Touch, the video-capable iPod Nano, and the compact iPod Shuffle. The iPhone can function as an iPod but is generally treated as a separate product. Former iPod models include the iPod Mini and the spin-off iPod Photo (since reintegrated into the main iPod Classic line). iPod Classic models store media on an internal hard drive, while all other models use flash memory to enable their smaller size (the discontinued Mini used a Microdrive miniature hard drive). As with many other digital music players, iPods can also serve as external data storage devices. Storage capacity varies by model.

Apple's iTunes software can be used to transfer music to the devices from computers using certain versions of Apple Macintosh and Microsoft Windows operating systems.[1] For users who choose not to use Apple's software or whose computers cannot run iTunes software, several open source alternatives to iTunes are also available.[2] iTunes and its alternatives may also transfer photos, videos, games, contact information, e-mail settings, Web bookmarks, and calendars to iPod models supporting those features. As of September 2008, more than 173,000,000 iPods had been sold worldwide, making it the best-selling digital audio player series in history.[3]

Contents

History and design



The iPod Classic 5G (right) and 6G (left) showing the improved album view

The iPod line came from Apple's "digital hub" category,[4] when the company began creating software for the growing market of personal digital devices. Digital cameras, camcorders and organizers had well-established mainstream markets, but the company found existing digital music players "big and clunky or small and useless" with user interfaces that were "unbelievably awful,"[4] so Apple decided to develop its own. As ordered by CEO Steve Jobs, Apple's hardware engineering chief Jon Rubinstein assembled a team of engineers to design the iPod line, including hardware engineers Tony Fadell and Michael Dhuey,[5] and design engineer Jonathan Ive.[4] The product was developed in less than one year and unveiled on 23 October 2001. Jobs announced it as a Mac-compatible product with a 5 GB hard drive that put "1,000 songs in your pocket."[6]

Apple did not develop the iPod software entirely in-house, instead using PortalPlayer's reference platform based on 2 ARM cores. The platform had rudimentary software running on a commercial microkernel embedded operating system. PortalPlayer had previously been working on an IBM-branded MP3 player with Bluetooth headphones.[4] Apple contracted another company, Pixo, to help design and implement the user interface under the direct supervision of Steve Jobs.[4] As development progressed, Apple continued to refine the software's look and feel. Starting with the iPod Mini, the Chicago font was replaced with Espy Sans. Later iPods switched fonts again to Podium Sans—a font similar to Apple's corporate font, Myriad. iPods with color displays then adopted some Mac OS X themes like Aqua progress bars, and brushed metal meant to evoke a combination lock. In 2007, Apple modified the iPod interface again with the introduction of the sixth-generation iPod Classic and third-generation iPod Nano by changing the font to Helvetica and, in most cases, splitting the screen in half by displaying the menus on the left and album artwork, photos, or videos on the right (whichever was appropriate for the selected item).

In September 2007, during the course of a lawsuit with patent holding company Burst.com, Apple drew attention to a patent for a similar device that was developed in 1979. Kane Kramer patented the idea of a "plastic music box" in 1979, which he called the IXI.[7] He was unable to secure funding to renew the US$ 120,000 worldwide patent, so it lapsed and Kramer never profited from his idea.[7] Kramer is now in talks with the company to discuss how he will be reimbursed.[7]

Trademark

The name *iPod* was proposed by Vinnie Chieco, a freelance copywriter, who (with others) was called by Apple to figure out how to introduce the new player to the public. After Chieco saw a prototype, he thought of the

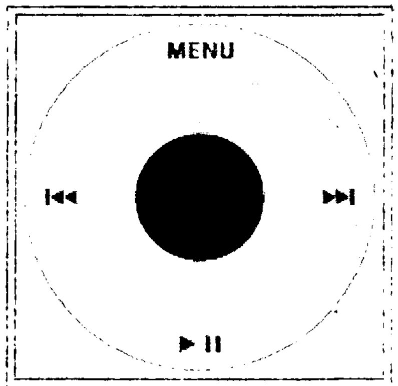
movie *2001: A Space Odyssey* and the phrase "Open the pod bay door, Hal!", which refers to the white EVA Pods of the Discovery One spaceship.[4] Apple researched the trademark and found that it was already in use. Joseph N. Grasso of New Jersey had originally listed an "iPod" trademark with the U.S. Patent and Trademark Office in July 2000 for Internet kiosks. The first iPod kiosks had been demonstrated to the public in New Jersey in March 1998, and commercial use began in January 2000, but had apparently been discontinued by 2001. The trademark was registered by the USPTO in November 2003, and Grasso assigned it to Apple Computer, Inc. in 2005.[8]

Software

The iPod line can play several audio file formats including MP3, AAC/M4A, Protected AAC, AIFF, WAV, Audible audiobook, and Apple Lossless. The iPod Photo introduced the ability to display JPEG, BMP, GIF, TIFF, and PNG image file formats. Fifth and sixth generation iPod Classics, as well as third generation iPod Nanos, can additionally play MPEG-4 (H.264/MPEG-4 AVC) and QuickTime video formats, with restrictions on video dimensions, encoding techniques and data-rates.[9] Originally, iPod software only worked with Mac OS; iPod software for Microsoft Windows was launched with the second generation model.[10] Unlike most other media players, Apple does not support Microsoft's WMA audio format—but a converter for WMA files without Digital Rights Management (DRM) is provided with the Windows version of iTunes. MIDI files also cannot be played, but can be converted to audio files using the "Advanced" menu in iTunes. Alternative open-source audio formats, such as Ogg Vorbis and FLAC, are not supported without installing custom firmware onto an iPod (e.g. Rockbox).

During installation, an iPod is associated with one host computer. Each time an iPod connects to its host computer, iTunes can synchronize entire music libraries or music playlists either automatically or manually. Song ratings can be set on an iPod and synchronized later to the iTunes library, and vice versa. A user can access, play, and add music on a second computer if an iPod is set to manual and not automatic sync, but anything added or edited will be reversed upon connecting and syncing with the main computer and its library. If a user wishes to automatically sync music with another computer, an iPod's library will be entirely wiped and replaced with the other computer's library.

User interface



The iPod line's signature click wheel

iPods with color displays use anti-aliased graphics and text, with sliding animations. All iPods (except the iPod shuffle and iPod touch) have five buttons and the later generations have the buttons integrated into the click wheel—an innovation that gives an uncluttered, minimalist interface. The buttons perform basic functions such as menu, play, pause, next track, and previous track. Other operations, such as scrolling through menu items and controlling the volume, are performed by using the click wheel in a rotational manner. The iPod shuffle does not have any controls on the actual player, instead it has a small control on the earphone cable, with volume-up and -down buttons and a single button for play/pause, next track, etc. The iPod Touch has no click-wheel. Instead it uses a 3.5" touch screen in addition to a home button, sleep/wake button and (on the second generation iPod touch) volume-up and -down buttons. The user interface for the iPod touch is virtually identical to that of the iPhone. Both devices use the iPhone OS.

iTunes Store

The iTunes Store is an online media store run by Apple and accessed via iTunes. It was introduced on 29 April 2003 and it sells individual songs, with typical prices being US$0.99, or $1.29 AU$1.69 (inc. GST), NZ$1.79 (inc. GST), €0.99 (inc. VAT), or £0.79 (inc. VAT) per song. Since no other portable player supports the DRM used, only iPods can play protected content from the iTunes Store. The store became the market leader soon after its launch[11] and Apple announced the sale of videos through the store on 12 October 2005. Full-length movies became available on 12 September 2006.[12]

Purchased audio files use the AAC format with added encryption. The encryption is based on the FairPlay DRM system. Up to five authorized computers and an unlimited number of iPods can play the files. Burning the files onto an audio CD, then re-compressing can create music files without the DRM, although this results in reduced quality. The DRM can also be removed using third-party software. However, in a deal with Apple, EMI began selling DRM-free, higher-quality songs on the iTunes Stores, in a category called "iTunes Plus." While individual songs were made available at a cost of US$1.29, 30¢ more than the cost of a regular DRM song, entire albums were available for the same price, US$9.99, as DRM encoded albums. On 17 October 2007, Apple lowered the cost of individual iTunes Plus songs to US$0.99 per song, the same as DRM encoded tracks. On January 6, 2009, Apple announced that DRM has been removed from 80% of the music catalog, and that it will be removed from all music by April, 2009.

iPods cannot play music files from competing music stores that use rival-DRM technologies like Microsoft's protected WMA or RealNetworks' Helix DRM. Example stores include Napster and MSN Music. RealNetworks claims that Apple is creating problems for itself[13] by using FairPlay to lock users into using the iTunes Store. Steve Jobs has stated that Apple makes little profit from song sales, although Apple uses the store to promote iPod sales.[14] However, iPods can also play music files from online stores that do not use DRM, such as eMusic or Amie Street.

Universal Music Group decided not to renew their contract with the iTunes Music Store on 3 July 2007. Universal will now supply iTunes in an 'at will' capacity.[15]

Apple debuted the iTunes Wi-Fi Music Store on 5 September 2007, in its Media Event entitled "The Beat Goes On..." This service allows users to access the Music Store from either an iPhone or an iPod Touch and download songs directly to the device that can be synced to the user's iTunes Library.

Games

Video games are playable on various versions of iPods. The original iPod had the game *Brick* (originally invented by Apple's co-founder Steve Wozniak) included as an easter egg hidden feature; later firmware versions added it as a menu option. Later revisions of the iPod added three more games in addition to *Brick*: *Parachute*, *Solitaire*, and *Music Quiz*.

In September 2006 the iTunes Store began to offer additional games for purchase with the launch of iTunes 7, compatible with the fifth generation iPod with iPod software 1.2 or later. Those games were: *Bejeweled*, *Cubis 2*, *Mahjong*, *Mini Golf*, *Pac-Man*, *Tetris*, *Texas Hold 'Em*, *Vortex*, and *Zuma*. Additional games have since been added. These games work on current and immediate previous generation of the iPod Nano and iPod Classic.

With third parties like Namco, Square Enix, Electronic Arts, Sega, and Hudson Soft all making games for the iPod, Apple's MP3 player has taken great steps towards entering the video game handheld console market. Even video game magazines like GamePro and EGM have reviewed and rated most of their games as of late.

The games are in the form of .ipg files, which are actually .zip archives in disguise. When unzipped, they reveal

executable files along with common audio and image files, leading to the possibility of third party games. Apple has not publicly released a software development kit (SDK) for iPod-specific development.[16] Apps produced with the iPhone SDK are compatible only with the iPhone OS on the iPod Touch and iPhone, which cannot run clickwheel-based games.

File storage and transfer

All iPods except for the iPod Touch can function in "disk mode" as a mass storage devices to store data files.[17] If an iPod is formatted on a Mac OS X computer it uses the HFS+ file system format, which allows it to serve as a boot disk for a Mac computer.[18] If it is formatted on Windows, the FAT32 format is used. With the advent of the Windows-compatible iPod, the default file system used on the iPod line switched from HFS+ to FAT32, although it can be reformatted to either file system (excluding the iPod Shuffle which is strictly FAT32). Generally, if a new iPod (excluding the iPod Shuffle) is initially plugged into a computer running Windows, it will be formatted with FAT32, and if initially plugged into a Mac running Mac OS X it will be formatted with HFS+.

Unlike many other MP3 players, simply copying audio or video files to the drive with a typical file management application will not allow an iPod to properly access them. The user must use software that has been specifically designed to transfer media files to iPods, so that the files are playable and viewable. Usually iTunes is used to transfer media to an iPod, though several alternative third-party applications are available on a number of different platforms.

iTunes 7 and above can transfer purchased media of the iTunes Store from an iPod to a computer, provided that computer containing the DRM protected media is authorized to play it.

Media files are stored on an iPod in a hidden folder, along with a proprietary database file. The hidden content can be accessed on the host operating system by enabling hidden files to be shown. The media files can then be recovered manually by copying the files or folders off the iPod. Many third-party applications also allow easy copying of media files off of an iPod.

Hardware

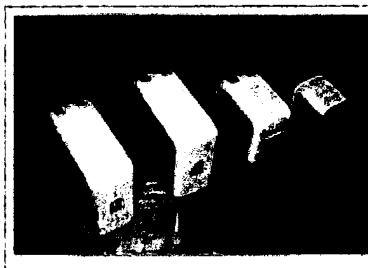
Chipsets and Electronics

Chipset or Electronic	Product(s)	Component(s)
Microcontroller	iPod Classic first to third generations	Two ARM 7TDMI-derived CPUs running at 90 MHz
	iPod Classic fourth and fifth generations, iPod Mini, iPod Nano first generation	Variable-speed ARM 7TDMI CPUs, running at a peak of 80 MHz to save battery life
	iPod Nano second generation	Samsung System-On-Chip, based around an ARM processor.[19]
	iPod Shuffle first generation	SigmaTel STMP3550 chip that handles both the music decoding and the audio circuitry.[20]
Audio Chip	All iPods (except the iPod Shuffle, 6G Classic and 2G Touch) [21]	Audio Codecs developed by Wolfson Microelectronics
	Sixth generation iPod Classic	Cirrus Logic Audio Codec Chip

Storage Medium	iPod Classic	45.7 mm (1.8 in) hard drives (ATA-6, 4200 rpm with proprietary connectors) made by Toshiba
	iPod Mini	25.4 mm (1 in) Microdrive by Hitachi and Seagate
	iPod Nano	Flash Memory from Samsung, Toshiba, and others
	iPod shuffle and Touch	Flash Memory
Batteries	iPod Classic first and second generation, Shuffle	Internal Lithium Polymer Batteries
	iPod Classic 3G onward, iPod Mini, iPod Nano, iPod Touch,	Internal Lithium-Ion Batteries

Connectivity

Originally, a FireWire connection to the host computer was used to update songs or recharge the battery. The battery could also be charged with a power adapter that was included with the first four generations. The third generation began including a 30-pin dock connector, allowing for FireWire or USB connectivity. This provided better compatibility with non-Apple machines, as most of them did not have FireWire ports at the time. Eventually Apple began shipping iPods with USB cables instead of FireWire, although the latter was available separately. As of the first generation iPod Nano and the fifth generation iPod Classic, Apple discontinued using FireWire for data transfer (while still allowing for use of FireWire to charge the device) in an attempt to reduce cost and form factor. As of the second-generation iPod Touch and the fourth-generation iPod Nano, FireWire charging ability has been removed. The second and third generation iPod Shuffle uses a single 3.5 mm jack which acts as both a headphone jack and a data port for the dock.



Four iPod wall chargers, with FireWire (left) and USB (right three) connectors, which allow iPods to charge without a computer. Notice how the units have been miniaturized.

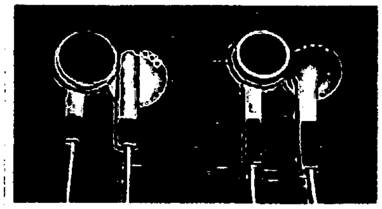
The dock connector also allowed the iPod to connect to accessories, which often supplement the iPod's music, video, and photo playback. Apple sells a few accessories, such as the now-discontinued iPod Hi-Fi, but most are manufactured by third parties such as Belkin and Griffin. Some peripherals use their own interface, while others use the iPod's own screen. Because the dock connector is a proprietary interface, the implementation of the interface requires paying royalties to Apple.[22]

Accessories

Many accessories have been made for the iPod line. A large number are made by third party companies, although many, such as the late iPod Hi-Fi, are made by Apple. This market is sometimes described as the iPod ecosystem.[23] Some accessories add extra features that other music players have, such as sound recorders, FM radio tuners, wired remote controls, and audio/visual cables for TV connections. Other accessories offer unique features like the Nike+iPod pedometer and the iPod Camera Connector. Other notable accessories include external speakers, wireless remote controls, protective cases/films and wireless earphones.[24] Among the first accessory manufacturers were Griffin Technology, Belkin, JBL, Bose, Monster Cable, and SendStation.

BMW released the first iPod automobile interface,[25] allowing drivers of newer BMW vehicles to control an iPod using either the built-in steering

wheel controls or the radio head-unit buttons. Apple announced in 2005 that similar systems would be available for other vehicle brands, including Mercedes-Benz,[26] Volvo,[27] Nissan, Toyota,[28] Alfa Romeo, Ferrari,[29] Acura, Audi, Honda,[30] Renault, Infiniti[31] and Volkswagen.[32] Scion offers standard iPod connectivity on all their cars.



Two designs of iPod earphones. The current version is shown on the right.

Some independent stereo manufacturers including JVC, Pioneer, Kenwood, Alpine, Sony, and Harman Kardon also have iPod-specific integration solutions. Alternative connection methods include adaptor kits (that use the cassette deck or the CD changer port), audio input jacks, and FM transmitters such as the iTrip—although personal FM transmitters are illegal in some countries. Many car manufacturers have added audio input jacks as standard.[33]

Beginning in mid-2007, four major airlines, United, Continental, Delta, and Emirates reached agreements to install iPod seat connections. The free service will allow passengers to power and charge an iPod, and view video and music libraries on individual seat-back displays.[34] Originally KLM and Air France were reported to be part of the deal with Apple, but they later released statements explaining that they were only contemplating the possibility of incorporating such systems.[35]

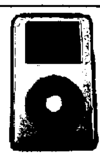
Audio performance

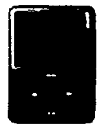
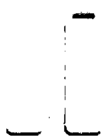
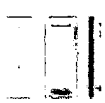
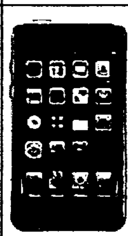
The third generation iPod had a weak bass response, as shown in audio tests.[36][37] The combination of the undersized DC-blocking capacitors and the typical low-impedance of most consumer headphones form a high-pass filter, which attenuates the low-frequency bass output. Similar capacitors were used in the fourth generation iPods.[38] The problem is reduced when using high-impedance headphones and is completely masked when driving high-impedance (line level) loads, such as an external headphone amplifier. The first generation iPod Shuffle uses a dual-transistor output stage,[36] rather than a single capacitor-coupled output, and does not exhibit reduced bass response for any load.

From the 5th generation iPod on, Apple introduced a user-configurable volume limit in response to concerns about hearing loss.[39] Users report that in the 6th generation iPod, the maximum volume output level is limited to 100dB in EU markets. Apple previously had to remove iPods from shelves in France.[40]

Models

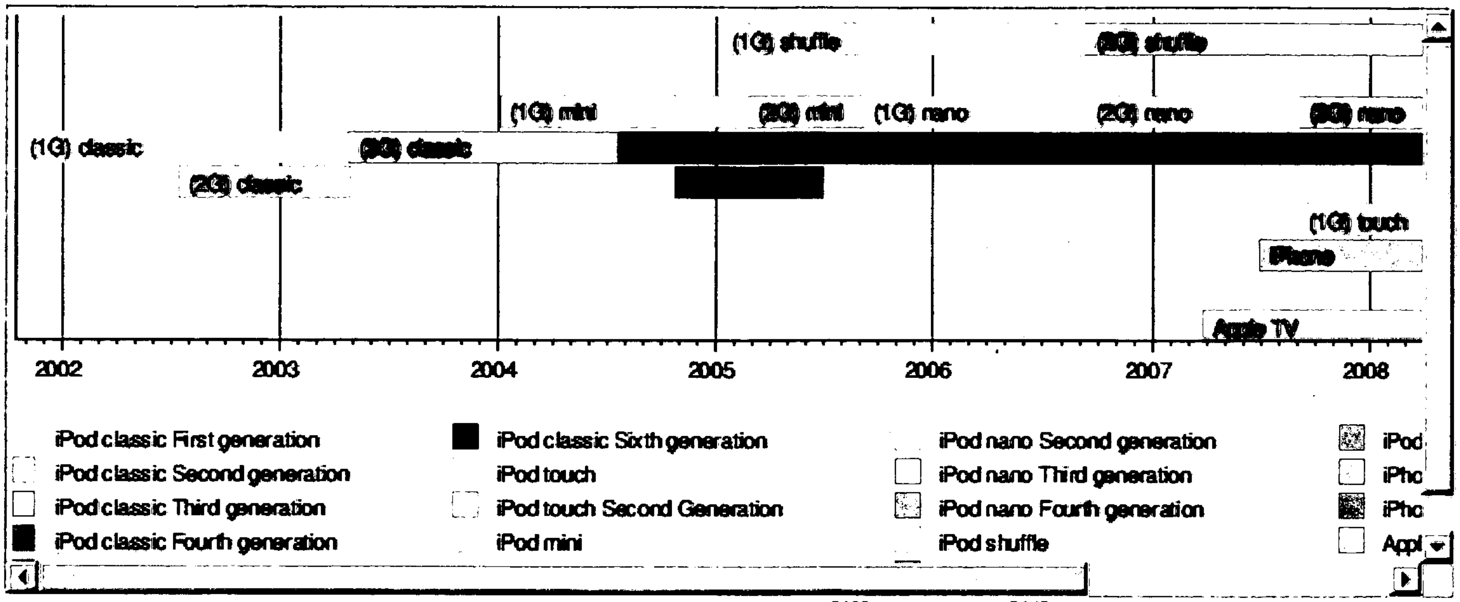
Model	Generation	Image	Capacity	Connection	Original release date	Minimum OS to sync	Rated battery life (hours)
	first		5, 10 GB	FireWire	23 October 2001	Mac: 9, 10.1	*audio*: 10
			First model, with mechanical scroll wheel. 10 GB model released later.				
	second		10, 20 GB	FireWire	17 July 2002	Mac: 10.1 Win: 2000	*audio*: 10
			Touch-sensitive wheel. FireWire port had a cover. Hold switch revised. Windows compatibility through Musicmatch.				
			10, 15,	FireWire	28 April	Mac: 10.1	

<table>
<tr><td rowspan="11">Classic</td><td rowspan="2">third</td><td rowspan="2"></td><td>20, 30, 40 GB</td><td>(USB for syncing only)</td><td>2003</td><td>Win: 2000</td><td colspan="3">audio: 8</td></tr>
<tr><td colspan="7">First complete redesign with all-touch interface, dock connector, and slimmer case. Musicmatch support dropped with later release of iTunes 4.1 for Windows.</td></tr>
<tr><td rowspan="5">fourth (Photo) (Color)</td><td rowspan="2"></td><td>20, 40 GB</td><td>FireWire or USB</td><td>19 July 2004</td><td>Mac: 10.2
Win: 2000</td><td colspan="3">audio: 12</td></tr>
<tr><td colspan="7">Adopted Click Wheel from iPod Mini, hold switch redesigned.</td></tr>
<tr><td rowspan="3"></td><td>photo: 30, 40, 60 GB</td><td rowspan="2">FireWire or USB</td><td>26 October 2004</td><td rowspan="2">Mac: 10.2
Win: 2000</td><td rowspan="2" colspan="3">audio: 15
slideshow: 5</td></tr>
<tr><td>color: 20, 60 GB</td><td>28 June 2005</td></tr>
<tr><td colspan="7">Premium spin-off of 4G iPod with color screen and picture viewing. Later re-integrated into main iPod line.</td></tr>
<tr><td rowspan="2">fifth</td><td rowspan="2"></td><td>30, 60, 80 GB</td><td>USB (FireWire for charging only)</td><td>12 October 2005</td><td>Mac: 10.3
Win: 2000</td><td>30 GB
audio: 14
video: 2 (later 3.5)</td><td colspan="2">60/80 GB
audio: 20
video: 3/6.5</td></tr>
<tr><td colspan="7">Second full redesign with a slimmer case, and larger screen with video playback. Offered in black or white. Hardware and firmware updated with 60 GB model replaced with 80 GB model on 12 September 2006.</td></tr>
<tr><td rowspan="2">sixth</td><td rowspan="2"></td><td>80, 120, 160 GB</td><td>USB (FireWire for charging only)</td><td>5 September 2007</td><td>Mac: 10.4
Win: XP</td><td>80 GB
audio: 30
video: 5</td><td>120 GB
audio: 36
video: 6</td><td>160 GB
audio: 40
video: 7</td></tr>
<tr><td colspan="7">Introduced the "Classic" suffix. New interface and anodized aluminum front plate. Silver replaces white. In September 2008 the hardware and firmware was updated with a 120 GB model replacing the 80 GB model. The 160 GB model was discontinued.</td></tr>
<tr><td rowspan="4">Mini</td><td rowspan="2">first</td><td rowspan="2"></td><td>4 GB</td><td>USB or FireWire</td><td>6 January 2004</td><td>Mac: 10.1
Win: 2000</td><td colspan="3">audio: 8</td></tr>
<tr><td colspan="7">New smaller model, available in 5 colors. Introduced the "Click Wheel".</td></tr>
<tr><td rowspan="2">second</td><td rowspan="2"></td><td>4, 6 GB</td><td>USB or FireWire</td><td>22 February 2005</td><td>Mac: 10.2
Win: 2000</td><td colspan="3">audio: 18</td></tr>
<tr><td colspan="7">Brighter color variants with longer battery life. Click Wheel lettering matched body color. Gold color discontinued. Later replaced by iPod Nano.</td></tr>
<tr><td rowspan="2"></td><td rowspan="2">first</td><td rowspan="2"></td><td>1, 2, 4 GB</td><td>USB (FireWire for charging only)</td><td>7 September 2005</td><td>Mac: 10.3
Win: 2000</td><td colspan="3">audio: 14
slideshow: 4</td></tr>
<tr><td colspan="7">Replaced Mini. Available in black or white and used flash memory. Color screen for picture viewing. 1 GB version released later.</td></tr>
</table>

Nano	**second**		2, 4, 8 GB	USB (FireWire for charging only)	12 September 2006	Mac: 10.3 Win: 2000	*audio*: 24 *slideshow*: 5
			Anodized aluminum casing and 6 colors available.				
	third		4, 8 GB	USB (FireWire for charging only)	5 September 2007	Mac: 10.4 Win: XP	*audio*: 24 *video*: 5
			2" QVGA screen, colors refreshed with chrome back, new interface, video capability, smaller Click Wheel.				
	fourth		4, 8, 16 GB	USB	9 September 2008	Mac: 10.4 Win: XP	*audio*: 24 *video*: 4
			Revert to tall form and all-aluminum enclosure with 9 color choices, added accelerometer for shake and horizontal viewing. 4 GB model limited release in select markets.				
shuffle	**first**		512 MB, 1 GB	USB (no adaptor required)	11 January 2005	Mac: 10.2 Win: 2000	*audio*: 12
			New entry-level model. Uses flash memory and has no screen.				
	second		1, 2 GB	USB	12 September 2006	Mac: 10.3 Win: 2000	*audio*: 12
			Smaller clip design with anodized aluminum casing. 4 color options added later. Colors were later refreshed twice.				
	third		4 GB	USB	11 March 2009	Mac: 10.4 Win: XP	*audio*: 10
			Smaller design with controls relocated to right earbud cable. Two colors, and features VoiceOver.				
Touch	**first**		8, 16, 32 GB	USB (FireWire for charging only)	5 September 2007	Mac: 10.4 Win: XP	*audio*: 22 *video*: 5
			First iPod with Wi-Fi and a Multi-Touch interface. Features Safari browser and wireless access to the iTunes Store and YouTube. 32 GB model later added. iPhone OS 2.0 and App Store access requires an upgrade fee.				
	second		8, 16, 32 GB	USB	9 September 2008	Mac: 10.4 Win: XP	*audio*: 36 *video*: 6
			New tapered chrome back with Nike+ functionality, volume buttons, and built-in speaker added. iPhone OS 2.0 and App Store access standard. Bluetooth support added but not made active until iPhone OS 3.0.				
Sources: Apple Inc.[41], *Mactracker*[42]							

Timeline of iPod models

See also: Template:Timeline of full-size iPod models, Template:Timeline of compact iPod models, and Timeline of Apple products



Sources: Apple press release library,[43] *Mactracker*[44]

Patent disputes

In 2005, Apple faced two lawsuits claiming patent infringement by the iPod line and its associated technologies:[45] Advanced Audio Devices claimed the iPod line breached its patent on a "music jukebox",[46] while a Hong Kong-based IP portfolio company called Pat-rights filed a suit claiming that Apple's FairPlay technology breached a patent[47] issued to inventor Ho Keung Tse. The latter case also includes the online music stores of Sony, RealNetworks, Napster, and Musicmatch as defendants.[48]

Apple's application to the United States Patent and Trademark Office for a patent on "rotational user inputs",[49] as used on the iPod interface, received a third "non-final rejection" (NFR) in August 2005. Also in August 2005, Creative Technology, one of Apple's main rivals in the MP3 player market, announced that it held a patent[50] on part of the music selection interface used by the iPod line, which Creative dubbed the "Zen Patent", granted on 9 August 2005.[51] On 15 May 2006, Creative filed another suit against Apple with the United States District Court for the Northern District of California. Creative also asked the United States International Trade Commission to investigate whether Apple was breaching U.S. trade laws by importing iPods into the United States.[52]

On 24 August 2006, Apple and Creative announced a broad settlement to end their legal disputes. Apple will pay Creative US$100 million for a paid-up license, to use Creative's awarded patent in all Apple products. As part of the agreement, Apple will recoup part of its payment, if Creative is successful in licensing the patent. Creative then announced its intention to produce iPod accessories by joining the *Made for iPod* program.[53]

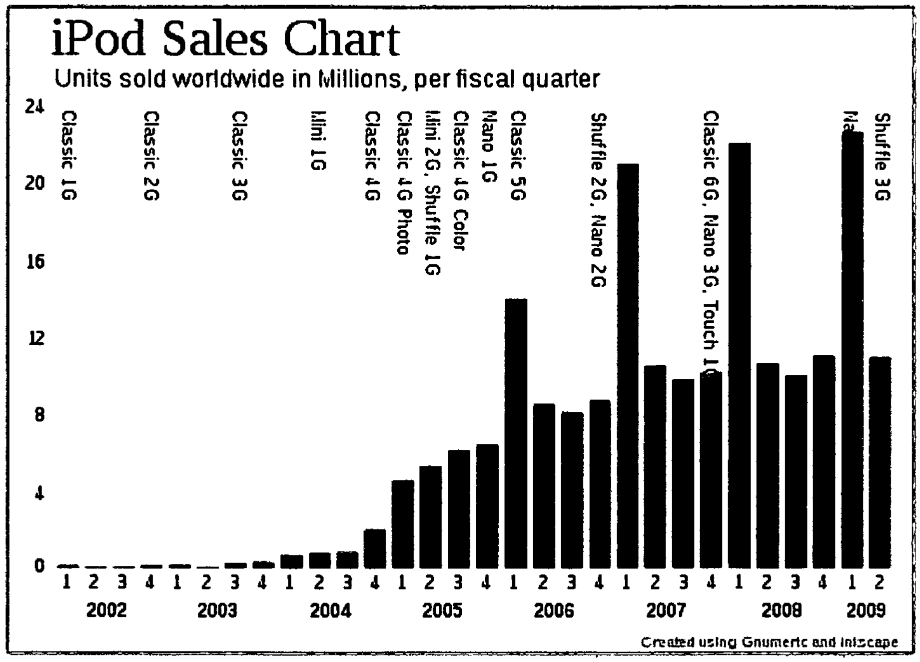
Sales

See also: iPod advertising

Since October 2004, the iPod line has dominated digital music player sales in the United States, with over 90% of the market for hard drive-based players and over 70% of the market for all types of players.[54] During the year from January 2004 to January 2005, the high rate of sales caused its U.S. market share to increase from

31% to 65% and in July 2005, this market share was measured at 74%. In January 2007 the iPod market share reached 72.7% according to Bloomberg Online.

The release of the iPod Mini helped to ensure this success at a time when competing flash-based music players were once dominant.[55] On 8 January 2004, Hewlett-Packard (HP) announced that they would sell HP-branded iPods under a license agreement from Apple. Several new retail channels were used—including Wal-Mart—and these iPods eventually made up 5% of all iPod sales. In July 2005, HP stopped selling iPods due to unfavorable terms and conditions imposed by Apple.[56]



iPod quarterly sales. Click for table of data and sources. Note that Q1 is October through December of previous year, the holiday season.

In January 2007, Apple reported record quarterly revenue of US$7.1 billion, of which 48% was made from iPod sales.[57]

On 9 April 2007, it was announced that Apple had sold its one-hundred millionth iPod, making it the biggest selling digital music player of all time. In April 2007, Apple reported second quarter revenue of US$5.2 billion, of which 32% was made from iPod sales.[58] Apple and several industry analysts suggest that iPod users are likely to purchase other Apple products such as Mac computers.[59]

On 5 September 2007, during their "The Beat Goes On" event, Apple announced that the iPod line had surpassed 110 million units sold.

On 22 October 2007, Apple reported quarterly revenue of US$6.22 billion, of which 30.69% came from Apple notebook sales, 19.22% from desktop sales and 26% from iPod sales. Apple's 2007 year revenue increased to US$24.01 billion with US$3.5 billion in profits. Apple ended the fiscal year 2007 with US$15.4 billion in cash and no debt.[60]

On 22 January 2008, Apple reported the best quarter revenue and earnings in Apple's history so far. Apple posted record revenue of US$9.6 billion and record net quarterly profit of US$1.58 billion. 42% of Apple's revenue for the First fiscal quarter of 2008 came from iPod sales, followed by 21% from notebook sales and 16% from desktop sales.[61]

On 21 October 2008, Apple reported that only 14.21% of total revenue for fiscal quarter 4 of year 2008 came from iPods.[62] Apple has sold over 206 million iPods to date (see chart).

Industry impact

iPods have won several awards ranging from engineering excellence,[63] to most innovative audio product,[64] to fourth best computer product of 2006.[65] iPods often receive favorable reviews; scoring on looks, clean

design, and ease of use. PC World says that iPod line has "altered the landscape for portable audio players".[64] Several industries are modifying their products to work better with both the iPod line and the AAC audio format. Examples include CD copy-protection schemes,[66] and mobile phones, such as phones from Sony Ericsson and Nokia, which play AAC files rather than WMA.

In addition to its reputation as a respected entertainment device, iPods have also become accepted as business devices. Government departments, major institutions and international organisations have turned to the iPod line as a delivery mechanism for business communication and training, such as the Royal and Western Infirmaries in Glasgow, Scotland, where iPods are used to train new staff.[67]

iPods have also gained popularity for use in education. Apple offers more information on educational uses for iPods on their website,[68] including a collection of lesson plans. There has also been academic research done in this area in nursing education[69] and more general K-16 education.[70] Duke University provided iPods to all incoming freshmen in the fall of 2004, and the iPod program continues today with modifications.[71]

Criticism

Battery issues

The advertised battery life on most models is different from the real-world achievable life. For example, the fifth generation 30 GB iPod is advertised as having up to 14 hours of music playback. An MP3.com report stated that this was virtually unachievable under real-life usage conditions, with a writer for MP3.com getting on average less than 8 hours from an iPod.[72] In 2003, class action lawsuits were brought against Apple complaining that the battery charges lasted for shorter lengths of time than stated and that the battery degraded over time.[73] The lawsuits were settled by offering individuals either US$50 store credit or a free battery replacement.[74]

iPod batteries are not designed to be removed or replaced by the user, although some users have been able to open the case themselves, usually following instructions from third-party vendors of iPod replacement batteries. Compounding the problem, Apple initially would not replace worn-out batteries. The official policy was that the customer should buy a refurbished replacement iPod, at a cost almost equivalent to a brand new one. All lithium-ion batteries eventually lose capacity during their lifetime[75] (guidelines are available for prolonging life-span) and this situation led to a market for third-party battery replacement kits.

Apple announced a battery replacement program on 14 November 2003, a week before[76] a high publicity stunt and website by the Neistat Brothers.[77] The initial cost was US$99,[78] and it was lowered to US$59 in 2005. One week later, Apple offered an extended iPod warranty for US$59.[79] For the iPod Nano, soldering tools are needed because the battery is soldered onto the main board. Fifth generation iPods have their battery attached to the backplate with adhesive.[80][81]

Reliability and durability

iPods have been criticized for their short life-span and fragile hard drives. A 2005 survey conducted on the MacInTouch website found that the iPod line had an average failure rate of 13.7% (although they note that comments from respondants indicate that "the true iPod failure rate may be lower than it appears"). It concluded that some models were more durable than others.[82] In particular, failure rates for iPods employing hard drives was usually above 20% while those with flash memory had a failure rate below 10%, indicating poor hard drive durability. In late 2005, many users complained that the surface of the first generation iPod Nano can become scratched easily, rendering the screen unusable.[83][84] A class action lawsuit was also filed.[85] Apple initially

considered the issue a minor defect, but later began shipping these iPods with protective sleeves.

Allegations of worker exploitation

On 11 June 2006, the British newspaper *Mail on Sunday* reported that iPods are mainly manufactured by workers who earn no more than US$50 per month and work 15-hour shifts.[86] Apple investigated the case with independent auditors and found that, while some of the plant's labour practices met Apple's Code of Conduct, others did not: Employees worked over 60 hours a week for 35% of the time, and worked more than six consecutive days for 25% of the time.[87]

Foxconn, Apple's manufacturer, initially denied the abuses,[88] but when an auditing team from Apple found that workers had been working longer hours than were allowed under Chinese law, they promised to prevent workers working more hours than the code allowed. Apple hired a workplace standards auditing company, Verité, and joined the Electronic Industry Code of Conduct Implementation Group to oversee the measures. On 31 December 2006, workers at the Longhua, Shenzhen factory (owned by Foxconn) formed a union. The union is affiliated with the world's largest and most powerful federation of trade unions, the All-China Federation of Trade Unions.[89]

Social isolation

Tara Brabazon, professor of media studies at the University of Brighton, is concerned that iPods may cause social isolation.[90] A school in Sydney, Australia banned MP3 players to encourage students to communicate with others.[91]

See also

- Comparison of portable media players
- Comparison of iPod managers

References

1. ^ Apple Inc.. "iTunes system requirements. Apple iTunes software currently runs on Macintosh OS X 10.3.9 or OS X 10.4.9 or later and on Microsoft Windows XP (Service Pack 2) or Vista (http://www.apple.com/itunes/download/)". http://www.apple.com/itunes/download/. Retrieved on 2008-05-28.
2. ^ Ross McKillop, simplehelp.net. "Alternatives to iTunes for managing your iPod (http://www.simplehelp.net/2007/07/08/10-alternatives-to-itunes-for-managing-your-ipod/)". http://www.simplehelp.net/2007/07/08/10-alternatives-to-itunes-for-managing-your-ipod/. Retrieved on 2008-05-28.
3. ^ Ryan Block (5 September 2007). "Steve Jobs live -- Apple's "The beat goes on" special event (http://www.engadget.com/2007/09/05/steve-jobs-live-apples-the-beat-goes-on-special-event/17)". *Engadget*. http://www.engadget.com/2007/09/05/steve-jobs-live-apples-the-beat-goes-on-special-event/17. Retrieved on 2008-03-10.
4. ^ *a b c d e f* Kahney, Leander.Straight Dope on the iPod's Birth (http://www.wired.com/news/columns/cultofmac/0,71956-0.html), *Wired News*, 2006-10-17. Retrieved on 2006-10-30.
5. ^ "2007 Engineer of the Year Finalist Michael Dhuey's Hardware Knowledge Helps Breathe Life Into iPod, TelePresence (http://www.designnews.com/article/CA6478178.html)", *Design News*, 24 September 2007.
6. ^ http://lowendmac.com/orchard/05/origin-of-the-ipod.html
7. ^ *a b c* Boffey, Daniel Apple admit Briton DID invent iPod, but he's still not getting any money (http://www.dailymail.co.uk/news/article-1053152/Apple-admit-Briton-DID-invent-iPod-hes-getting-money.html) *Daily Mail*, 2008-09-08. Retrieved on 2008-09-08.
8. ^ Serial No. 78018061, Registration No. 2781793, records of the U.S. Patent and Trademark Office. InPub, LLC, filed an "IPOD" trademark on 1 June 1999, for "computer software and hardware." The trademark was abandoned

May 18, 2000, without commercial use.
9. ^ iPod Classic Technical Specs (http://www.apple.com/ipodclassic/specs.html)
10. ^ iTunes Downloads (http://www.apple.com/itunes/download/)
11. ^ iTunes Music Store Catalog Tops One Million Songs (http://www.apple.com/pr/library/2004/aug/10itms.html), *Apple Inc.*, 2004-08-10. Retrieved on 2006-12-28.
12. ^ Scott-Joynt, Jeremy. Apple targets TV and film market (http://news.bbc.co.uk/1/hi/business/5339470.stm), *BBC News*, 2006-09-12. Retrieved on 2006-09-12.
13. ^ Kanellos, Michael. Real's Glaser exhorts Apple to open iPod (http://news.com.com/2100-1025_3-5177914.html), *CNet News*, 2004-03-23. Retrieved on 2006-06-20.
14. ^ Orlowski, Andrew. Your 99c belong to the RIAA – Steve Jobs (http://www.theregister.co.uk/2003/11/07/your_99c_belong/), *The Register*, 2003-11-07. Retrieved on2006-06-20.
15. ^ Evans, Jonny. Universal confirms iTunes contract change (http://www.macworld.co.uk/ipod-itunes/news/index.cfm?newsid=18459), *Macworld UK*, 2007-07-04. Retrieved on 2007-07-05.
16. ^ "What's Inside an iPod Game?" (http://www.bensinclair.com/article/whats-inside-an-ipod-game) bensinclair.com, September 14, 2006.
17. ^ iPod touch: Appears in iTunes but not in Finder or Windows Desktop (http://docs.info.apple.com/article.html?artnum=306414)
18. ^ How To: Boot up your Mac from your iPod (http://lifehacker.com/software/how-to/boot-up-your-mac-from-your-ipod-260157.php)
19. ^ Cassell, Jonathan. Apple Delivers More For Less With New iPod Nano (http://www.isuppli.com/markech/default.asp?id=360), iSuppli Corporation, 2006-09-20. Retrieved on 2006-10-21.
20. ^ Williams, Martyn. How Much Should an IPod Shuffle Cost? (http://www.pcworld.com/article/119799-1/article.html), *PC World*, 2005-02-24. Retrieved on 2006-08-14.
21. ^ Macworld Wolfson loses Apple iPod business (http://www.macworld.co.uk/ipod-itunes/news/index.cfm?RSS&NewsID=20818)
22. ^ Johnson, Joel (2008-07-10). "How the "Apple Tax" Boosts Prices on iPod & iPhone Accessories (http://www.popularmechanics.com/technology/industry/4272628.html?page=1)". Popular Mechanics. http://www.popularmechanics.com/technology/industry/4272628.html?page=1. Retrieved on 2008-09-09.
23. ^ Darlin, Damon. The iPod Ecosystem (http://www.nytimes.com/2006/02/03/technology/03ipod.html?ex=1296622800&en=91f4e87dd848693f&ei=5088). *New York Times*, 2006-02-03. Retrieved on 2006-08-14.
24. ^ In-The-Ear Bluetooth Earphones (http://www.etymotic.com/ephp/er88.aspx). Retrieved on 2007-02-17.
25. ^ iPod Your BMW (http://www.ipodyourbmw.com/). Retrieved on 2007-02-17.
26. ^ Apple & Mercedes-Benz Unveil iPod Integration Kit (http://www.apple.com/pr/library/2005/jan/11mercedes.html), *Apple Inc.*, 2005-01-11. Retrieved on 2006-06-20.
27. ^ Apple & Volvo Announce iPod Connectivity For Entire 2005 US Model Line (http://www.apple.com/pr/library/2005/jan/11volvo.html), *Apple Inc.*, 2005-01-11. Retrieved on 2006-06-20.
28. ^ http://www.gizmag.com/go/7945/
29. ^ Apple & Leading Car Companies Team Up to Deliver iPod Integration in 2005 (http://www.apple.com/pr/library/2005/jan/11cars.html), *Apple Inc.*, 2005-01-11. Retrieved on 2006-06-20.
30. ^ Honda Music Link for iPods (http://automobiles.honda.com/models/audio_accessories.asp?ModelName=Accord%20Coupe&PHOTO=0), *Honda*. Retrieved on 2007-02-17.
31. ^ Apple Car Integration page (http://www.apple.com/ipod/carintegration/#car_browser_infiniti)
32. ^ Apple Teams Up With Acura, Audi, Honda & Volkswagen to Deliver Seamless iPod Experience (http://www.apple.com/pr/library/2005/sep/07ipod_auto.html), *Apple Inc.*, 2005-09-07. Retrieved on 2006-06-20.
33. ^ Car Integration: iPod your car (http://www.apple.com/ipod/carintegration.html), *Apple Inc.*. Retrieved on 2007-02-17.
34. ^ Apple Teams Up With Continental, Delta, Emirates, & United to deliver iPod Integration (http://www.apple.com/pr/library/2006/nov/14ipod.html), *Apple Inc.*, 2006-11-14. Retrieved on 2006-12-07.
35. ^ Marsal, Katie. Two of six airlines say there's no ink on iPod deal (http://www.appleinsider.com/article.php?id=2240), *AppleInsider*, 2006-11-15. Retrieved on 2006-12-07.
36. ^ ***a b*** Machrone, Bill. iPod audio measurements (http://home.comcast.net./~machrone/playertest/playertest.htm), *PC Magazine*, 2005. Retrieved on 2007-02-17.
37. ^ Heijligers, Marc. iPod audio measurements (http://members.chello.nl/~m.heijligers/ipod/Performance/measurements.html). Retrieved on 2007-02-17.
38. ^ Heijligers, Marc. iPod circuit design engineering (http://homepage.mac.com/marc.heijligers/audio/ipod/engineering/engineering.html), May 2006. Retrieved on 2007-02-17.
39. ^ Cohen, Peter. iPod update limits iPod volume setting

(http://www.macworld.com/article/50101/2006/03/ipodupdate.html), Macworld, 2006. Retrieved on 2008-11-07.
40. ^ Fried, Ian. Apple pulls iPod in France (http://news.cnet.com/2100-1040-960211.html). Retrieved on 2008-11-07.
41. ^ Apple Inc., Identifying iPod models (http://support.apple.com/kb/HT1353), retrieved 31 October 2007.
42. ^ Mactracker, Apple Inc. model database, version as of 26 July 2007.
43. ^ Apple press release library (http://www.apple.com/pr/library/), Retrieved September 19, 2007.
44. ^ Mactracker, Apple Inc. model database, version as of 26 July 2007.
45. ^ Apple faces patent lawsuits over its iPod (http://www.channelregister.co.uk/2005/03/10/apple_ipod_patent_lawsuits/), *ChannelRegister*, 2005-03-10. Retrieved on 2007-02-17.
46. ^ U.S. Patent 6,587,403 (http://www.google.com/patents?vid=USPAT6587403&id=oawMAAAAEBAJ&dq=6,587,403) — Advanced Audio Devices' "music jukebox" patent.
47. ^ U.S. Patent 6,665,797 (http://www.google.com/patents?vid=USPAT6665797&id=QrR2AAAAEBAJ&dq=6,665,797) — "Protection of software again against unauthorized use" (corrected to "Computer Apparatus/Software Access Control").
48. ^ Apple, Sony among those named in new DRM lawsuit (http://www.appleinsider.com/article.php?id=1235), *AppleInsider*, 2005-08-16. Retrieved on 2007-02-17.
49. ^ U.S. patent application 20030095096 (http://appft1.uspto.gov/netacgi/nph-Parser?p=1&u=/netahtml/PTO/search-bool.html&r=1&f=G&l=50&co1=AND&d=PG01&s1=%2260/346,237%22&OS=%2260/346,237%22&RS=%2260/346,237%22) Apple Inc.'s application on "rotational user inputs".
50. ^ U.S. Patent 6,928,433 (http://www.google.com/patents?vid=USPAT6928433&id=sJoVAAAAEBAJ&dq=6,928,433) Creative Technology's "Zen" patent.
51. ^ Creative wins MP3 player patent (http://news.bbc.co.uk/2/hi/technology/4198360.stm), *BBC News*, 2005-08-30. Retrieved on 2007-02-17.
52. ^ Creative sues Apple over patent (http://www.macworld.co.uk/news/index.cfm?NewsID=14646), *Macworld UK*, 2006-05-16. Retrieved on 2007-03-20.
53. ^ Apple & Creative Announce Broad Settlement... (http://www.apple.com/pr/library/2006/aug/23settlement.html), , *Apple Inc.*. Retrieved on 2007-02-17.
54. ^ Marsal, Katie. iPod: how big can it get? (http://www.appleinsider.com/article.php?id=1770), *AppleInsider*, 2006-05-24. Retrieved on 2007-02-17.
55. ^ jomy Apple Computer (presentation slides) (http://homepage.mac.com/jomy/PhotoAlbum14.html), 2005-10-18. Retrieved on 2006-05-25.
56. ^ HP to stop selling Apple's iPods (http://www.appleinsider.com/articles/05/07/29/hp_to_stop_selling_apples_ipods.html), *AppleInsider*, 2005-07-29. Retrieved on 2007-08-06.
57. ^ Apple Reports First Quarter Results (http://www.apple.com/pr/library/2007/01/17results.html), *Apple Inc.*, 2007-01-17. Retrieved on 2007-02-17.
58. ^ Apple Reports Second Quarter Results (http://www.apple.com/pr/library/2007/04/25results.html), *Apple Inc.*. Retrieved on 2007-04-25.
59. ^ Orlowski, Andrew. For Apple, Halo effect eclipses Osborne effect (http://www.theregister.co.uk/2005/10/11/apple_q4_2005/), *The Register*, 2005-10-11. Retrieved on 2006-07-13.
60. ^ Apple Reports Fourth Quarter 2007 Results (http://www.apple.com/pr/library/2007/10/22results.html), *Apple Inc.*, 2007-10-22. Retrieved on 2007-10-22.
61. ^ Apple Inc. (22 January 2008). Apple Reports First Quarter Results (http://www.apple.com/pr/library/2008/01/22results.html). Press release. Retrieved on 2008-1-23
62. ^ AppleInsider (27 October 2008). [1] (http://www.appleinsider.com/articles/08/10/27/ipods_accounted_for_just_14_2_of_apples_sales_in_q4_2008.html). Retrieved on 2008-10-27
63. ^ iPod and Bluetooth lead to prizes (http://news.bbc.co.uk/1/hi/technology/4605881.stm), *BBC News*, 2005-06-03. Retrieved on 2007-03-20.
64. ^ ***a b*** "The 25 Most Innovative Products of the Year (http://www.pcworld.com/article/id,123942-page,2/article.html)". PC World. 2007-08-03. http://www.pcworld.com/article/id,123942-page,2/article.html. Retrieved on 2007-02-17.
65. ^ Apple wins 5 'World Class' awards (http://www.macnn.com/articles/06/05/31/top.100.products.announced/), *MacNN*. Retrieved on 2007-02-17.
66. ^ Apple, iPod, and CD Copy Protection (http://www.macrumors.com/pages/2004/07/20040720124549.shtml), *MacRumors*. Retrieved on 2007-02-17.
67. ^ Hospitals train staff with iPods (http://news.bbc.co.uk/1/hi/business/4859302.stm), *BBC News*, 2006-03-29. Retrieved on 2007-06-16.
68. ^ Mobile Learning (http://www.apple.com/education/teachers-professors/mobile-learning.html)

69. ^ Maag, M.E.D. (2006). "Podcasting and MP3 Players: Emerging Education Technologies. (http://www.cinjournal.com/pt/re/cin/fulltext.00024665-200601000-00005.htm)". *CIN: Computers, Informatics, Nursing* **24** (1): 9–13. doi:10.1186/1742-5581-3-1 (http://dx.doi.org/10.1186/1742-5581-3-1). http://www.cinjournal.com/pt/re/cin/fulltext.00024665-200601000-00005.htm. Retrieved on 2008-02-08.
70. ^ Slykhuis, D. (2006). *Have an iPod? Then you need to know this about how to use it in your classroom. (http://www.aace.org/newdl/index.cfm?fuseaction=Reader.ViewAbstract&paper_id=22435).* http://www.aace.org/newdl/index.cfm?fuseaction=Reader.ViewAbstract&paper_id=22435. Retrieved on 2008-02-08.
71. ^ Duke's Ever-Evolving iPod Initiative (http://www.insidehighered.com/news/2006/04/28/ipod)
72. ^ MP3 Insider: The truth about your battery life (http://www.mp3.com/features/stories/3646.html), *mp3.com*, 2006-03-13. Retrieved on 2006-07-10.
73. ^ Apple investigates iPod batteries (http://news.bbc.co.uk/1/hi/technology/3477497.stm), *BBC News*, 2004-02-10. Retrieved on 2007-03-20.
74. ^ Horwitz, Jeremy. Apple's iPod Battery Settlement, Explained (http://www.ilounge.com/index.php/articles/comments/apples-ipod-battery-settlement-explained/), *iLounge*, 2005-06-10. Retrieved on 2006-08-27.
75. ^ The Curse of Lithium Ion Batteries (http://www.mp3newswire.net/stories/6002/li_ion.html), *MP3 Newswire*, 2006-01-06. Retrieved on 2006-11-30.
76. ^ iPod Battery FAQ (http://ipodbatteryfaq.com/#18). Retrieved on 2006-11-26.
77. ^ Neistat, Casey. A Message From the Neistat Brothers (http://www.ipodsdirtysecret.com/message.html), 2003-11-20. Retrieved on 2007-02-17.
78. ^ Apple offers iPod battery replacement service (http://www.macminute.com/2003/11/14/ipodbattery), *MacMinute*, 2003-11-14. Retrieved on 2006-11-26.
79. ^ AppleCare for iPod now available (http://www.macminute.com/2003/11/21/ipodapplecare/), *MacMinute*, 2003-11-21. Retrieved on 2006-11-26.
80. ^ Ecker, Clint. Vivisection of the Video iPod (http://arstechnica.com/reviews/hardware/video-ipod.ars/6), *Ars Technica*, 2005-10-19. Retrieved on 2006-11-26.
81. ^ Disassemble Guide for Video iPod (http://www.ipodmods.com/guides/Video/videodisassemblyguide.htm). Retrieved on 2006-11-26.
82. ^ iPod Reliability Survey (http://www.macintouch.com/reliability/ipodfailures.html), *MacInTouch*, 2005-11-28. Retrieved on 2006-10-29.
83. ^ Apple responds to iPod nano screen concerns (http://www.macworld.com/news/2005/09/27/nanoscreen/index.php), *Macworld*, 2005-09-27. Retrieved on 2007-02-17.
84. ^ Arthur, Charles. iPod Nano owners in screen scratch trauma (http://www.theregister.co.uk/2005/09/23/ipod_nano_scratching/), *The Register*, 2005-09-25. Retrieved on 2007-02-17.
85. ^ Fried, Ina. Suit filed over Nano scratches (http://news.com.com/Suit+filed+over+Nano+scratches/2100-1047_3-5906399.html), *CNet News*, 2005-10-21. Retrieved on 2007-02-17.
86. ^ Inside Apple's iPod factories (http://www.macworld.co.uk/news/index.cfm?NewsID=14915), *Macworld UK*, 2006-06-12. Retrieved on 2007-03-20.
87. ^ Millard, Elizabeth. Is It Ethical To Own an iPod? (http://www.newsfactor.com/story.xhtml?story_id=13100EV85O19&page=2). Retrieved on 2007-03-20.
88. ^ Foxconn denies iPod 'sweatshop' claims (http://www.macnn.com/articles/06/06/19/ipod.sweatshop.claims/), *MacNN*, 2006-06-19. Retrieved on 2007-02-17.
89. ^ Bodeen, Chistopher. Official Union Forms at Taiwanese Plant (http://www.forbes.com/feeds/ap/2007/01/17/ap3334400.html), *Forbes*, 2007-01-17. Retrieved on 2007-02-17.
90. ^ http://www.timeshighereducation.co.uk/story.asp?storyCode=401340&sectioncode=26
91. ^ http://www.news24.com/News24/Technology/News/0,,2-13-1443_1679706,00.html

External links

- Apple iPod (http://www.apple.com/itunes/) — Official website
- iPod troubleshooting basics and service FAQ (http://docs.info.apple.com/article.html?artnum=32222) — From the official website
- *Apple's 21st century Walkman* (http://money.cnn.com/magazines/fortune/fortune_archive/2001/11/12/313342/index.htm) — Brent Schlender, *Fortune*, October 2001

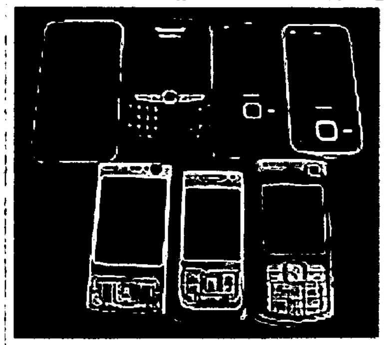
- iPod Nation (http://www.newsweek.com/id/54529) — Steven Levy, *Newsweek*, July 2004
- The Perfect Thing (http://www.wired.com/wired/archive/14.11/ipod.html) — Steven Levy, *Wired*, November 2006

Retrieved from "http://en.wikipedia.org/wiki/IPod"
Categories: 2001 introductions | Portable media players | Industrial design examples | IPod | IPhone OS software | ITunes
Hidden categories: Articles containing potentially dated statements from September 2008 | All articles containing potentially dated statements | All pages needing cleanup | Articles with minor POV problems from January 2009 | All articles with unsourced statements | Articles with unsourced statements from January 2009 | Articles with unsourced statements from December 2008 | Cleanup from section

- This page was last modified on 28 July 2009 at 23:48.

Smartphone

From Wikipedia, the free encyclopedia

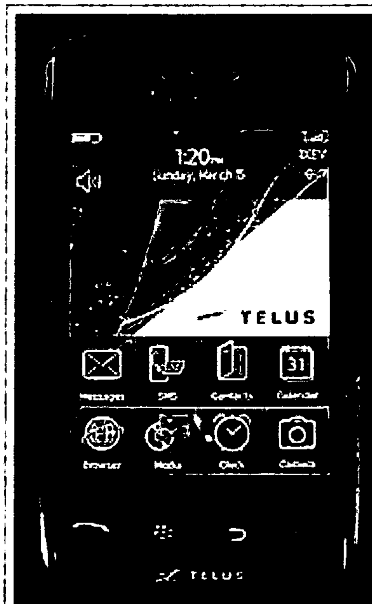


A collection of smartphones. From left to right, top row: iPhone 3G, Blackberry 8820, Nokia N78, Nokia N81, (bottom row) Nokia N95, Nokia E65, Nokia N70.

A **smartphone** is a mobile phone offering advanced capabilities, often with PC-like functionality (PC-mobile handset convergence). There is no industry standard definition of a smartphone.[1][2] For some, a smartphone is a phone that runs complete operating system software providing a standardized interface and platform for application developers.[3] For others, a smartphone is simply a phone with advanced features like e-mail, Internet and e-book reader capabilities, and/or a built-in full keyboard or external USB keyboard and VGA connector. In other words, it is a miniature computer that has phone capability.[4][5]

Growth in demand for advanced mobile devices boasting powerful processors, abundant memory, large screens and open operating systems has outpaced the rest of the mobile phone market for several years.[6]

Contents



The BlackBerry Storm, made by Research In Motion

Definition

There is no agreement in the industry about what precisely constitutes a "smart" phone, and definitions have changed over time.[2] According to David Wood, EVP at Symbian Ltd., "Smart phones differ from ordinary mobile phones in two fundamental ways: how they are built and what they can do."[7] Other definitions put different stresses on these two factors...

"With smart phones it's just one evolution in one evolution, so it might be that the actual device at some point ... will become even smaller and we will not call it a phone anymore, but it will be integrated ... the deal here is to make the device as invisible as possible, between you, and what you want to do," says Sacha Wunsch-Vincent at the OECD.[8]

Most devices considered smartphones today use an identifiable operating system, often with the ability to add applications (e.g. for enhanced data processing, connectivity or entertainment) - in contrast to regular phones which only support sandboxed applications (like Java games). These smartphone applications may be developed by the manufacturer of the device, by the network operator or by any other third-party software developer, since the operating system is open..

In terms of features, most smartphones support full featured email capabilities with the functionality of a

complete personal organizer. Other functionality might include an additional interface such as a miniature QWERTY keyboard, a touch screen or a D-pad, a built-in camera, contact management, an accelerometer, built-in navigation hardware and software, the ability to read business documents in a variety of formats such as PDF and Microsoft Office, media software for playing music, browsing photos and viewing video clips, internet browsers or even just secure access to company mail, such as is provided by a BlackBerry. One common feature to the majority of the smartphones is a contact list able to store as many contacts as the available memory permits, in contrast to regular phones that has a limit to the maximum number of contacts that can be stored.

The iPhone, made by Apple Inc.

History

The first smartphone was called Simon; it was designed by IBM in 1992 and shown as a concept product[9] that year at COMDEX, the computer industry trade show held in Las Vegas, Nevada. It was released to the public in 1993 and sold by BellSouth. Besides being a mobile phone, it also contained a calendar, address book, world clock, calculator, note pad, e-mail, send and receive fax, and games. It had no physical buttons to dial with. Instead customers used a touch-screen to select phone numbers with a finger or create facsimiles and memos with an optional stylus. Text was entered with a unique on-screen "predictive" keyboard. By today's standards, the Simon would be a fairly low-end product, however its feature set at the time was incredibly advanced.

The Nokia E71 smartphone running S60 3rd Edition, Feature Pack 1 UI on the Symbian OS v9.2

The Nokia Communicator line was the first of Nokia's smartphones starting with the Nokia 9000, released in 1996. This distinctive palmtop computer style smartphone was the result of a collaborative effort of an early successful and expensive PDA model by Hewlett Packard combined with Nokia's bestselling phone around that time and early prototype models had the two devices fixed via a hinge; the Nokia 9210 as the first color screen Communicator model which was the first true smartphone with an open operating system; the 9500 Communicator that was also Nokia's first cameraphone Communicator and Nokia's first WiFi phone; the 9300 Communicator was the third dimensional shift into a smaller form factor; and the latest E90 Communicator includes GPS. The Nokia Communicator model is remarkable also having been the most expensive phone model sold by a major brand for almost the full lifespan of the model series, easily 20% and sometimes 40% more expensive than the next most expensive smartphone by any major manufacturer.

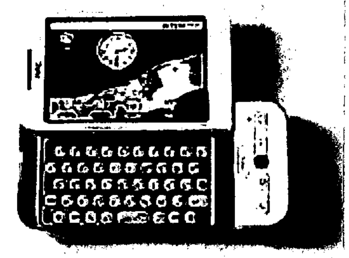
The Ericsson R380 was sold as a 'smartphone' but could not run native third-party applications.[10] Although the Nokia 9210 was arguably the first true smartphone with an open operating system, Nokia continued to refer to it as a Communicator.

In 2001 RIM released the first BlackBerry which was the first smartphone optimized for wireless email use and has achieved a total customer base of 8 million subscribers by June 2007, of which three quarters are in North America.

Although the Nokia 7650, announced in 2001, was referred to as a 'smart phone' in the media, and is now called a 'smartphone' on the Nokia support site, the press release referred to it as an 'imaging phone'.[11][12][13] Handspring delivered the first widely popular smartphone devices in the US market by marrying its Palm OS based Visor PDA together with a piggybacked GSM phone module, the VisorPhone. By 2002, Handspring was marketing an integrated smartphone called the Treo; the company subsequently merged with Palm primarily because the PDA market was dying but the Treo smartphone was quickly becoming popular as a phone with extended PDA organizer features. That same year, Microsoft announced its Windows CE Pocket PC OS would be offered as "Microsoft Windows Powered Smartphone 2002".[14] Microsoft originally defined its Windows Smartphone products as lacking a touchscreen and offering a lower screen resolution compared to its sibling Pocket PC devices. Palm has since largely abandoned its own Palm OS in favor of licensing Microsoft's WinCE-based operating system now referred to as Windows Mobile.

The Palm Treo Pro, a Windows Mobile smartphone

In 2005 Nokia launched its N-Series of 3G smartphones which Nokia started to market not as mobile phones but as multimedia computers.



The HTC Dream, a smartphone running the Android operating system

Out of 1 billion camera phones to be shipped in 2008, smartphones, the higher end of the market with full email support, will represent about 10% of the market or about 100 million units.

The Smartphone Summit semi-annual conference details smartphone industry market data, trends, and updates among smartphone related hardware, software, and accessories.

Android, a cross platform OS for smartphones was released in 2008. Android is an Open Source platform backed by Google, along with major hardware and software developers (such as Intel, HTC, ARM, and eBay, to name a few), that form the Open Handset Alliance[15].

The Palm Pre, a smartphone running the Palm webOS.

The first phone to use the Android OS is the HTC Dream, branded for distribution by T-Mobile as the G1[16]. The phone features a full, capacitive touch screen, a flip out QWERTY keyboard, and a track ball for navigating web pages. The software suite included on the phone consists of integration with Google's proprietary applications, such as Maps, Calendar, and Gmail, as well as Google's Chrome Lite full HTML web browser[17]. Third party apps are available for free via the Android Market, with premium apps slated for Q1 2009[18].

In July 2008 Apple introduced its innovative App Store with both for fee and free applications. The app store can deliver smartphone applications developed by third parties directly to the iPhone or iPod Touch over wifi or cellular network without using a PC to download. The App Store has been a huge success for Apple and by June 2009 hosted more than 50,000 applications.[19] The app store hit one billion application downloads on April 23, 2009.[20]

Following the popularity of Apple's App Store, many other mobile platforms are following Apple with their own application stores. Palm, Microsoft and Nokia have all announced they will launch Apple-like app stores. RIM recently launched its app store, BlackBerry App World.

Operating systems

Operating systems that can be found on smartphones include Symbian OS, iPhone OS, RIM's BlackBerry, Windows Mobile, Linux, Palm WebOS and Android.

The most common operating systems (OS) used in smartphones by Q4 2008 sales are:

Symbian OS from Symbian Ltd. (47.1% Market Share Sales Q4 2008 [21])
: Symbian has the largest share in most markets worldwide, but lags behind other companies in the relatively small but highly visible North American market.[22] This matches the success of its largest shareholder[23] and customer, Nokia, in all markets except Japan. Nokia itself enjoys 52.9% of the smartphone market.[24] In Japan Symbian is strong due to a relationship with NTT DoCoMo, with only one of the 44 Symbian handsets released in Japan coming from Nokia.[25] It is used by many major handset manufacturers, including BenQ, LG, Motorola, Samsung, and Sony Ericsson.[26] Various implementations of user interfaces on top of Symbian (most notable being UIQ and Nokia's own S60) are incompatible, which along with the requirement that applications running on mobile phones be signed[27] is hindering the potential for a truly widely accepted mobile application platform. It has received some adverse press attention due to virus threats (namely trojan horses).[28]

RIM BlackBerry operating system (19.5% Market Share Sales Q4 2008)
: This OS is focused on easy operation and was originally designed for business. Recently it has seen a surge in third-party applications and has been improved to offer full multimedia support.

Windows Mobile from Microsoft (12.4% Market Share Sales Q4 2008)
: The Windows CE operating system and Windows Mobile middleware are widely spread in Asia. The two improved variants of this operating system, Windows Mobile 6 Professional (for touch screen devices) and Windows Mobile 6 Standard, were unveiled in February 2007. Windows Mobile benefits from the low barrier to entry for third-party developers to write new applications for the platform. It has been criticized for having a user interface which is not optimized for touch input by fingers; instead, it is more usable with a stylus. However, unlike iPhone OS, it does support both touch screen and physical keyboard configurations.

iPhone OS from Apple Inc. (10.7% Market Share Sales Q4 2008)
: The iPhone uses an operating system called iPhone OS, which is derived from Mac OS X. Third party applications were not officially supported until the release of iPhone OS 2.0 on July 11th 2008. Before this,"jailbreaking" allowed third party applications to be installed, and this method is still available.

Linux operating system (8.4% Market Share Sales Q4 2008)
: Linux is strongest in China where it is used by Motorola, and in Japan, used by DoCoMo.[29][30] Rather than being a platform in its own right, Linux is used as a basis for a number of different platforms developed by several vendors, including Android, LiMo, Maemo, Openmoko and Qt Extended, which are mostly incompatible.[31][32] PalmSource (now Access) is moving towards an interface running on Linux.[33] Another platform based on Linux is being developed by Motorola, NEC, NTT DoCoMo, Panasonic, Samsung, and Vodafone.[34]

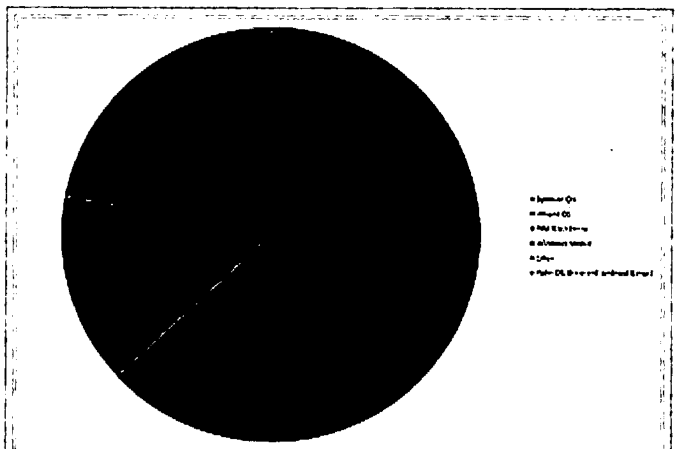
Palm webOS from Palm Inc. and Palm OS/Garnet OS from Access Co. (0.9% Market Share Sales Q4 2008)

Palm webOS is Palm's next generation operating system.[35] PalmSource traditionally used its own platform developed by Palm Inc. Access Linux Platform (ALP) is an improvement that was planned to be launched in the first half of 2007. It will use technical specifications from the Linux Phone Standards Forum. The Access Linux Platform will include an emulation layer to support applications developed for Palm-based devices.

Android from Google Inc. (Released 22 Oct 2008)

Android was developed by Google Inc.. Its share of the smartphone market is still small because of its recent release date. Android is an Open Source, Linux-derived platform backed by Google, along with major hardware and software developers (such as Intel, HTC, ARM, and eBay, to name a few), that form the Open Handset Alliance.[36] This OS, though very new, already has a cult following among programmers eager to develop apps for its flexible, Open Source, back end.[37] Android promises to give developers access to every aspect of the phone's operation.[38] This lends many to foresee the promise of further growth for the Android platform.[39]

Marketshare data from Gartner report "Worldwide Smartphone Sales Reached Its Lowest Growth Rate With 3.7 Per Cent Increase in Fourth Quarter of 2008"[21]



Market share of Smartphone operating systems (as of November 2008)

See also

- Android (mobile device platform)
- Comparison of smartphones
- Blackberry thumb
- Camera phone
- e-book reader
- Energy harvesting
- Flexible keyboard
- Information appliance
- mAh, to measure battery capacity
- Microbrowser
- Memory card
- PC-mobile handset convergence
- Text-to-speech
- Videophone

References

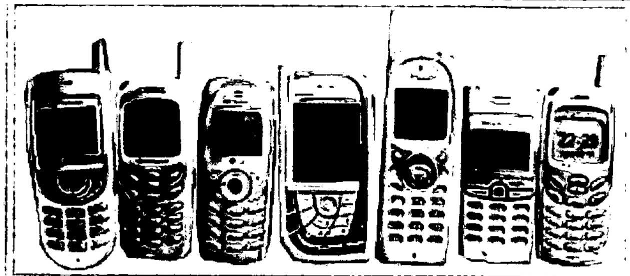
1. ^ Smartphones Take Center Stage at *Wi-fi Planet* (http://www.wi-fiplanet.com/news/article.php/3551686)
2. ^ *a b* Analysis: What is a smart phone? at *Silicon.com* (http://networks.silicon.com/mobile/0,39024665,39156391,00.htm)
3. ^ The (http://www.engadget.com/2007/01/09/the-iphone-is-not-a-smartphone/) iPhone is not a smartphone at *Engadget*
4. ^ CEVA Glossary of Terms (http://ceva-dsp.mediaroom.com/index.php?s=glossary)
5. ^ http://www.mytelus.com/ncp_news/article.en.do?pn=tech&articleID=2973940
6. ^ "Smart phones: how to stay clever in downturn (http://www.deloitte.co.uk/TMTPredictions/telecommunications/Smartphones-clever-in-downturn.cfm)". *Deloitte Telecommunications Predictions*. http://www.deloitte.co.uk/TMTPredictions/telecommunications/Smartphones-clever-in-downturn.cfm.
7. ^ *ibid* (http://networks.silicon.com/mobile/0,39024665,39156391,00.htm)
8. ^ Radio France International - Smartphones: Changing the way we communicate (http://telechargement.rfi.fr.edgesuite.net/rfi/anglais/audio/modules/actuen/R108/rfi_smartphone_feature_daniel_finna
9. ^ Schneidawind, J: "Big Blue unveiling", *USA Today*, November 23, 1992, page 2B
10. ^ Ericsson R380 (http://www.ericsson.com/solutions/enterprise/products/r380.shtml)
11. ^ Nokia Support Site (http://europe.nokia.com/A4143428)
12. ^ Nokia 7650 (http://www.theregister.co.uk/2001/11/20/nokia_7650_smart_phone_shame/)
13. ^ Imaging Phone (http://press.nokia.com/PR/200206/864667_5.html)
14. ^ Windows Powered Smartphone (http://www.microsoft.com/presspass/features/2002/Jan02/01-08msces.mspx)
15. ^ http://www.openhandsetalliance.com/oha_members.html
16. ^ http://www.t-mobileg1.com/?WT.srch=1&WT.mc_id=273m1&WT.z=p137999826
17. ^ http://www.engadget.com/2008/09/23/t-mobile-g1-first-hands-on/
18. ^ http://www.engadget.com/2008/12/31/paid-apps-still-coming-to-android-market-in-q1-09-us-and-uk-ro/
19. ^ "State Of The iPhone Ecosystem: 40 Million Devices and 50,000 Apps (http://www.techcrunch.com/2009/06/08/40-million-iphones-and-ipod-touches-and-50000-apps/)". *TechCrunch*. techcrunch.com. 2009. http://www.techcrunch.com/2009/06/08/40-million-iphones-and-ipod-touches-and-50000-apps/. Retrieved on 2009-07-05.
20. ^ "Apple's App Store: 1 Billion Served (http://www.techcrunch.com/2009/04/23/apples-app-store-1-billion-served/)". *TechCrunch*. techcrunch.com. 2009. http://www.techcrunch.com/2009/04/23/apples-app-store-1-billion-served/. Retrieved on 2009-07-06.
21. ^ *a b* http://www.gartner.com/it/page.jsp?id=910112
22. ^ North American Market (http://uk.theinquirer.net/?article=35179)
23. ^ Symbian Shareholders (http://www.symbian.com/about/overview/ownership/ownership.html)
24. ^ "Highlights from the Canalys Q3 2008 research (http://www.canalys.com/pr/2008/r2008112.htm)". *Canalys*. 2008-02-05. http://www.canalys.com/pr/2008/r2008112.htm.
25. ^ Symbian Facts (http://www.symbian.com/about/fastfacts/fastfacts.html)
26. ^ Symbian Licensees (http://www.symbian.com/about/overview/licensees/licensees.html)
27. ^ Symbian Signed (https://www.symbiansigned.com/app/page)
28. ^ Trojan Horses on Symbian (http://www.vnunet.com/vnunet/news/2127090/mobile-virus-moves-level)
29. ^ Shipping Linux-based Phones at *Technology News Daily* (http://www.technologynewsdaily.com/node/3904)
30. ^ Microsoft Excluded from DoCoMo at *The Register* (http://www.theregister.co.uk/2004/11/26/microsoft_excluded_from_docomo/)
31. ^ Incompatibility in Mobile Linux at *OS News* (http://www.osnews.com/story.php/15040/Editorial-The-Chaos-of-Incompatibility-in-Mobile-Linux)
32. ^ Search Mobile Computing (http://searchmobilecomputing.techtarget.com/originalContent/0,289142,sid40_gci1218922,00.html)
33. ^ Running on Linux (http://news.com.com/Can+Linux+save+the+Palm+OS/2008-1045_3-6110042.html)
34. ^ Motorola Press Release on Partnership on Linux Platform (http://www.motorola.com/mediacenter/news/detail.jsp?globalObjectId=6872_6826_23)
35. ^ Overview of webOS (http://developer.palm.com/webos_book/book1.html)
36. ^ http://www.openhandsetalliance.com/oha_members.html
37. ^ http://www.pcworld.com/businesscenter/article/146450/developers_praise_android_at_google_io.html?tk=rl_noinform

38. ^ http://code.google.com/android/
39. ^ http://www.engadget.com/2008/10/16/t-mobile-g1-review-part-2-software-and-wrap-up/

Retrieved from "http://en.wikipedia.org/wiki/Smartphone"
Categories: Smartphones | Touchscreens
Hidden categories: All articles with unsourced statements | Articles with unsourced statements from October 2007 | Articles with unsourced statements from February 2007 | Articles with unsourced statements from November 2008 | Articles needing additional references from July 2008

Mobile phone

From Wikipedia, the free encyclopedia



Several examples of non-flip mobile phones.

A **mobile phone** or **mobile** (also called **cellphone** and **handphone**,[1] as well as **cell phone, wireless phone, cellular phone, cell, cellular telephone, mobile telephone** or **cell telephone**) is a long-range, electronic device used for mobile voice or data communication over a network of specialized base stations known as cell sites. In addition to the standard voice function of a mobile phone, telephone, current mobile phones may support many additional services, and accessories, such as SMS for text messaging, email, packet switching for access to the Internet, gaming, Bluetooth, infrared, camera with video recorder and MMS for sending and receiving photos and video, MP3 player, radio and GPS. Most current mobile phones connect to a cellular network consisting of switching points and base stations (cell sites) owned by a mobile network operator (the exception is satellite phones, which are mobile but not cellular).

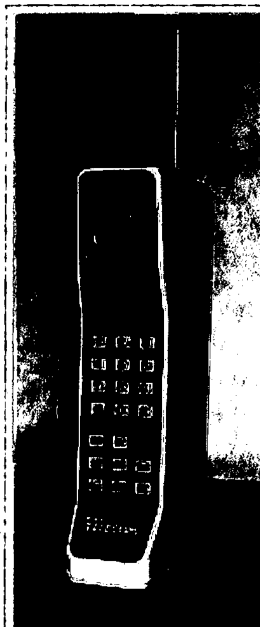
As opposed to a radio telephone, a mobile phone offers full duplex communication, automatised calling to and paging from a public switched telephone network (PSTN), handoff (am. English) or *handover* (European term) during a phone call when the user moves from one cell (base station coverage area) to another. A mobile phone offers wide area service, and should not be confused with a cordless telephone, which also is a **wireless phone**, but only offer telephony service within a limited range, e.g. within a home or an office, through a fixed line and a base station owned by the subscriber.

The International Telecommunication Union estimated that mobile cellular subscriptions worldwide would reach approximately 4.1 billion by the end of 2008.[2] Mobile phones have gained increased importance in the sector of Information and communication technologies for development in the 2000s and have effectively started to reach the bottom of the economic pyramid.[3]

Contents

History

In 1908, U.S. Patent 887,357 (http://www.google.com/patents?vid=887357) for a wireless telephone was issued in to Nathan B. Stubblefield of Murray, Kentucky. He applied this patent to "cave radio" telephones and not directly to cellular telephony as the term is currently understood.[4] Cells for mobile phone base stations were invented in 1947 by Bell Labs engineers at AT&T and further developed by Bell Labs during the 1960s. Radiophones have a long and varied history going back to Reginald Fessenden's invention and shore-to-ship demonstration of radio telephony, through the Second World War with military use of radio telephony links and civil services in the 1950s, while hand-held cellular radio devices have been available since 1973. A patent for the first wireless phone as we know today was issued in US Patent Number 3,449,750 (http://www.google.com/patents?id=sidyAAAAEBAJ&dq=george+sweigert) to George Sweigert of Euclid, Ohio on June 10, 1969.



Analog Motorola DynaTAC 8000X Advanced Mobile Phone System mobile phone as of 1983

In 1945, the zero generation (0G) of mobile telephones was introduced. Like other technologies of the time, it involved a single, powerful base station covering a wide area, and each telephone would effectively monopolize a channel over that whole area while in use. The concepts of frequency reuse and handoff, as well as a number of other concepts that formed the basis of modern cell phone technology, were described in the 1970s; see for example Fluhr and Nussbaum,[5] Hachenburg et. al.[6] , and U.S. Patent 4,152,647 (http://www.google.com/patents?vid=4152647), issued May 1, 1979 to Charles A. Gladden and Martin H. Parelman, both of Las Vegas, Nevada and assigned by them to the United States Government.

Martin Cooper, a Motorola researcher and executive is widely considered to be the inventor of the first practical mobile phone for hand-held use in a non-vehicle setting. Cooper is the first inventor named on "Radio telephone system" filed on October 17, 1973 with the US Patent Office and later issued as US Patent 3,906,166;[7] other named contributors on the patent included Cooper's boss, John F. Mitchell, Motorola's chief of portable communication products, who successfully pushed Motorola to develop wireless communication products that would be small enough to use outside the home, office or automobile and participated in the design of the cellular phone.[8][9] Using a modern, if somewhat heavy portable handset, Cooper made the first call on a hand-held mobile phone on April 3, 1973 to a rival, Dr. Joel S. Engel of Bell Labs.[10]

The first commercial citywide cellular network was launched in Japan by NTT in 1979. Fully automatic cellular networks were first introduced in the early to mid 1980s (the 1G generation). The Nordic Mobile Telephone (NMT) system went online in Denmark, Finland, Norway and Sweden in 1981.[11]

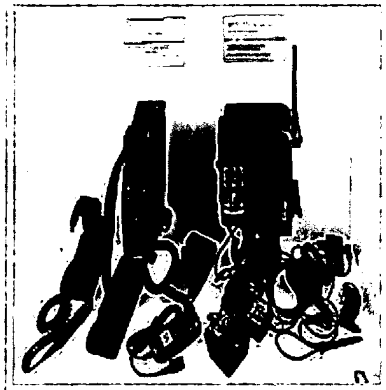
In 1983, Motorola DynaTAC was the first approved mobile phone by FCC in the United States. In 1984, Bell Labs developed modern commercial cellular technology (based, to a large extent, on the Gladden, Parelman Patent), which employed multiple, centrally controlled base stations (cell sites), each providing service to a small area (a cell). The cell sites would be set up such that cells partially overlapped. In a cellular system, a signal between a base station (cell site) and a terminal (phone) only need be strong enough to reach

Personal Handy-phone System mobiles and modems used in Japan around 1997–2003

between the two, so the same channel can be used simultaneously for separate conversations in different cells.

Cellular systems required several leaps of technology, including handover, which allowed a conversation to continue as a mobile phone traveled from cell to cell. This system included variable transmission power in both the base stations and the telephones (controlled by the base stations), which allowed range and cell size to vary. As the system expanded and neared capacity, the ability to reduce transmission power allowed new cells to be added, resulting in more, smaller cells and thus more capacity. The evidence of this growth can still be seen in the many older, tall cell site towers with no antennae on the upper parts of their towers. These sites originally created large cells, and so had their antennae mounted atop high towers; the towers were designed so that as the system expanded—and cell sizes shrank—the antennae could be lowered on their original masts to reduce range.

The first "modern" network technology on digital 2G (second generation) cellular technology was launched by Radiolinja (now part of Elisa Group) in 1991 in Finland on the GSM standard which also marked the introduction of competition in mobile telecoms when Radiolinja challenged incumbent Telecom Finland (now part of TeliaSonera) who ran a 1G NMT network.



A 1991 GSM mobile phone

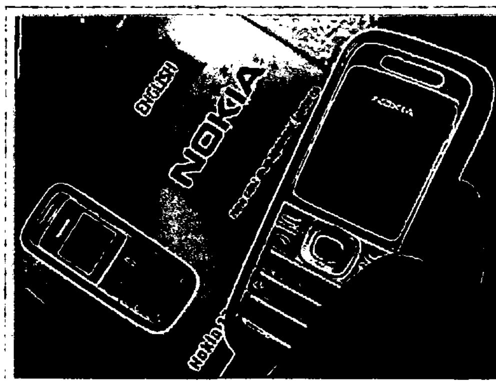
The first data services appeared on mobile phones starting with person-to-person SMS text messaging in Finland in 1993. First trial payments using a mobile phone to pay for a Coca Cola vending machine were set in Finland in 1998. The first commercial payments were mobile parking trialled in Sweden but first commercially launched in Norway in 1999. The first commercial payment system to mimic banks and credit cards was launched in the Philippines in 1999 simultaneously by mobile operators Globe and Smart. The first content sold to mobile phones was the ringing tone, first launched in 1998 in Finland. The first full internet service on mobile phones was i-Mode introduced by NTT DoCoMo in Japan in 1999.

In 2001 the first commercial launch of 3G (Third Generation) was again in Japan by NTT DoCoMo on the WCDMA standard.[12]

Until the early 1990s, following introduction of the Motorola MicroTAC, most mobile phones were too large to be carried in a jacket pocket, so they were typically installed in vehicles as car phones. With the miniaturization of digital components and the development of more sophisticated batteries, mobile phones have become smaller and lighter.

Handsets

There are several categories of mobile phones, from basic phones to feature phones such as musicphones and cameraphones, to smartphones. The first smartphone was the Nokia 9000 Communicator in 1996 which incorporated PDA functionality to the basic mobile phone at the time. As miniaturisation and increased processing power of microchips has enabled ever more features to be added to phones, the concept of the smartphone has evolved, and what was a high-end smartphone five years ago, is a standard phone today. Several phone series have been introduced to address a given market segment, such as the RIM BlackBerry focusing on



enterprise/corporate customer email needs; the SonyEricsson Walkman series of musicphones and Cybershot series of cameraphones; the Nokia N-Series of multimedia phones; and the Apple iPhone which provides full-featured web access and multimedia capabilities.

A Nokia phone with box.

Features

Mobile phones often have features beyond sending text messages and making voice calls, including call registers, GPS navigation, music (MP3) and video (MP4) playback, RDS radio receiver, alarms, memo and document recording, personal organiser and personal digital assistant functions, ability to watch streaming video or download video for later viewing, video calling, built-in cameras (1.0+ Mpx) and camcorders (video recording), with autofocus and flash, ringtones, games, PTT, memory card reader (SD), USB (2.0), infrared, Bluetooth (2.0) and WiFi connectivity, instant messaging, Internet e-mail and browsing and serving as a wireless modem for a PC, and soon will also serve as a console of sorts to online games and other high quality games.

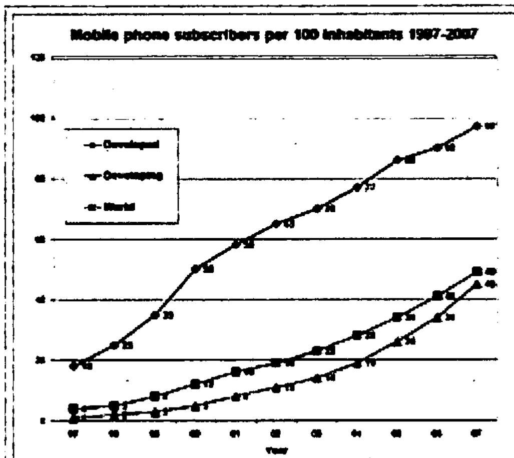
Some phones include a touchscreen.

The largest categories of mobile services are music, picture downloads, videogaming, adult entertainment, gambling, video/TV.

Nokia and the University of Cambridge are showing off a bendable cell phone called the Morph.[13]

Applications


A phone with touchscreen feature.


Mobile phone subscribers per 100 inhabitants 1997–2007

The most commonly used data application on mobile phones is SMS text messaging, with 74% of all mobile phone users as active users (over 2.4 billion out of 3.3 billion total subscribers at the end of 2007). SMS text messaging was worth over 100 billion dollars in annual revenues in 2007 and the worldwide average of messaging use is 2.6 SMS sent per day per person across the whole mobile phone subscriber base (source Informa 2007). The first SMS text message was sent from a computer to a mobile phone in 1992 in the UK, while the first person-to-person SMS from phone to phone was sent in Finland in 1993.

The other non-SMS data services used by mobile phones were worth 31 Billion dollars in 2007, and were led by mobile music, downloadable logos and pictures, gaming, gambling, adult entertainment and advertising (source:

Informa 2007). The first downloadable mobile content was sold to a mobile phone in Finland in 1998, when Radiolinja (now Elisa) introduced the downloadable ringing tone service. In 1999 Japanese mobile operator NTT DoCoMo introduced its mobile internet service, i-Mode, which today is the world's largest mobile internet service and roughly the same size as Google in annual revenues.

The first mobile news service, delivered via SMS, was launched in Finland in 2000. Mobile news services are expanding with many organisations providing "on-demand" news services by SMS. Some also provide "instant" news pushed out by SMS. Mobile telephony also facilitates activism and public journalism being explored by Reuters and Yahoo![14] and small independent news companies such as Jasmine News in Sri Lanka.

Companies like Monster.com are starting to offer mobile services such as job search and career advice. Consumer applications are on the rise and include everything from information guides on local activities and events to mobile coupons and discount offers one can use to save money on purchases. Even tools for creating websites for mobile phones are increasingly becoming available.

Mobile payments were first trialled in Finland in 1998 when two Coca-Cola vending machines in Espoo were enabled to work with SMS payments. Eventually the idea spread and in 1999 the Philippines launched the first commercial mobile payments systems, on the mobile operators Globe and Smart. Today mobile payments ranging from mobile banking to mobile credit cards to mobile commerce are very widely used in Asia and Africa, and in selected European markets. For example in the Philippines it is not unusual to have one's entire paycheck paid to the mobile account. In Kenya the limit of money transfers from one mobile banking account to another is one million US dollars. In India paying utility bills with mobile gains a 5% discount. In Estonia the government found criminals collecting cash parking fees, so the government declared that only mobile payments via SMS were valid for parking and today all parking fees in Estonia are handled via mobile and the crime involved in the activity has vanished.

Mobile Applications are developed using the Six M's (previously Five M's) service-development theory created by the author Tomi Ahonen with Joe Barrett of Nokia and Paul Golding of Motorola. The Six M's are Movement (location), Moment (time), Me (personalization), Multi-user (community), Money (payments) and Machines (automation). The Six M's / Five M's theory is widely referenced in the telecoms applications literature and used by most major industry players. The first book to discuss the theory was *Services for UMTS* by Ahonen & Barrett in 2002.

Power supply

Mobile phone charging service in Uganda

Mobile phones generally obtain power from batteries, which can be recharged from a USB port, from portable batteries, from mains power or a cigarette lighter socket in a car using an adapter (often called battery charger or wall wart) or from a solar panel or a dynamo (that can also use a USB port to plug the phone).

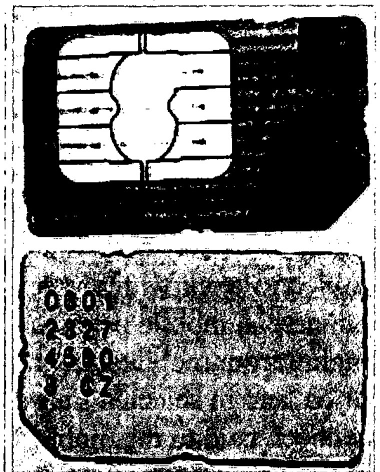
On 17 February 2009, the GSM Association announced[15] that they had agreed on a standard charger for mobile phones. The standard connector to be adopted by 17 manufacturers including Nokia, Motorola and Samsung is to be the micro-USB connector (several media reports erroneously reported this as the mini-USB). The new chargers will be much more efficient than existing chargers. Having a standard charger for all phones, means that manufacturers will no longer have to supply a charger with every new phone.

Formerly, the most common form of mobile phone batteries were nickel metal-hydride, as they have a low size

and weight. Lithium-Ion batteries are sometimes used, as they are lighter and do not have the voltage depression that nickel metal-hydride batteries do. Many mobile phone manufacturers have now switched to using lithium-Polymer batteries as opposed to the older Lithium-Ion, the main advantages of this being even lower weight and the possibility to make the battery a shape other than strict cuboid. Mobile phone manufacturers have been experimenting with alternative power sources, including solar cells.

SIM card

In addition to the battery, GSM mobile phones require a small microchip, called a Subscriber Identity Module or SIM Card, to function. Approximately the size of a small postage stamp, the SIM Card is usually placed underneath the battery in the rear of the unit, and (when properly activated) stores the phone's configuration data, and information about the phone itself, such as which calling plan the subscriber is using. When the subscriber removes the SIM Card, it can be re-inserted into another phone and used as normal.



Typical mobile phone SIM card

Each SIM Card is activated by use of a unique numerical identifier; once activated, the identifier is locked down and the card is permanently locked in to the activating network. For this reason, most retailers refuse to accept the return of an activated SIM Card.

Those cell phones that do not use a SIM Card have the data programmed in to their memory. This data is accessed by using a special digit sequence to access the "NAM" as in "Name" or number programming menu. From here, one can add information such as a new number for your phone, new Service Provider numbers, new emergency numbers, change their Authentication Key or A-Key code, and update their Preferred Roaming List or PRL. However, to prevent someone from accidentally disabling their phone or removing it from the network, the Service Provider puts a lock on this data called a Master Subsidiary Lock or MSL.

The MSL also ensures that the Service Provider gets payment for the phone that was purchased or "leased". For example, the Motorola RAZR V9C costs upwards of CAD $500. You can get one for approximately $200, depending on the carrier. The difference is paid by the customer in the form of a monthly bill. If the carrier did not use a MSL, then they may lose the $300–$400 difference that is paid in the monthly bill, since some customers would cancel their service and take the phone to another carrier.

The MSL applies to the SIM only so once the contract has been completed the MSL still applies to the SIM. The phone however, is also initially locked by the manufacturer into the Service Providers MSL. This lock may be disabled so that the phone can use other Service Providers SIM cards. Most phones purchased outside the US are unlocked phones because there are numerous Service Providers in close proximity to one another or have overlapping coverage. The cost to unlock a phone varies but is usually very cheap and is sometimes provided by independent phone vendors.

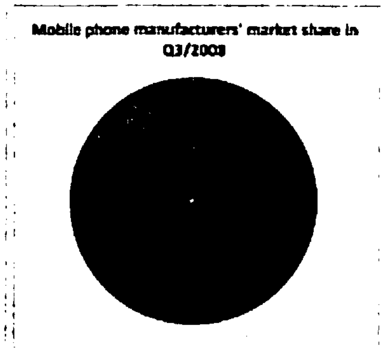
Having an unlocked phone is extremely useful for travelers due to the high cost of using the MSL Service Providers access when outside the normal coverage areas. It can cost sometimes up to 10 times as much to use a locked phone overseas as in the normal service area, even with discounted rates. T-Mobile will provide a SIM unlock code to account holders in good standing after 90 days according to their FAQ (https://support.t-mobile.com/doc/tm51885.xml?docid=3307).

For example, in Jamaica, an AT&T subscriber might pay in excess of US$1.65 per minute for discounted international service while a B-Mobile (Jamaican) customer would pay US$0.20 per minute for the same

international service. Some Service Providers focus sales on international sales while others focus on regional sales. For example, the same B-Mobile customer might pay more for local calls but less for international calls than a subscriber to the Jamaican national phone C&W (Cable & Wireless) company. These rate differences are mainly due to currency variations because SIM purchases are made in the local currency. In the US, this type of service competition does not exist because some of the major Service Providers do not offer Pay-As-You-Go services. [Needs Pay-As-You-Go references, rumored T-Mobile, Verizon provide one, AT&T does not as of 12/2008]

Market

In Q3/2008, Nokia was the world's largest manufacturer of mobile phones, with a global device market share of 39.4%, followed by Samsung (17.3%), Sony Ericsson (8.6%), Motorola (8.5%) and LG Electronics (7.7%). These manufacturers accounted for over 80% of all mobile phones sold at that time. [16]



Mobile phone manufacturers' market share in Q3/2008

Other manufacturers include Apple Inc., Audiovox (now UTStarcom), Benefon, BenQ-Siemens, CECT, High Tech Computer Corporation (HTC), Fujitsu, Kyocera, Mitsubishi Electric, NEC, Neonode, Panasonic, Palm, Matsushita, Pantech Wireless Inc., Philips, Qualcomm Inc., Research in Motion Ltd. (RIM), Sagem, Sanyo, Sharp, Siemens, Sendo, Sierra Wireless, SK Teletech, T&A Alcatel, Huawei, Trium and Toshiba. There are also specialist communication systems related to (but distinct from) mobile phones.

Media

The mobile phone became a mass media channel in 1998 when the first ringtones were sold to mobile phones by Radiolinja in Finland. Soon other media content appeared such as news, videogames, jokes, horoscopes, TV content and advertising. In 2006 the total value of mobile phone paid media content exceeded internet paid media content and was worth 31 Billion dollars (source Informa 2007). The value of music on phones was worth 9.3 Billion dollars in 2007 and gaming was worth over 5 billion dollars in 2007.[17]

The mobile phone is often called the Fourth Screen (if counting cinema, TV and PC screens as the first three) or Third Screen (counting only TV and PC screens). It is also called the Seventh of the Mass Media (with Print, Recordings, Cinema, Radio, TV and Internet the first six). Most early content for mobile tended to be copies of legacy media, such as the banner advertisement or the TV news highlight video clip. Recently unique content for mobile has been emerging, from the ringing tones and ringback tones in music to "mobisodes," video content that has been produced exclusively for mobile phones.

The advent of media on the mobile phone has also produced the opportunity to identify and track Alpha Users or Hubs, the most influential members of any social community. AMF Ventures measured in 2007 the relative accuracy of three mass media, and found that audience measures on mobile were nine times more accurate than on the internet and 90 times more accurate than on TV.

Related systems

Car phone

A type of telephone permanently mounted in a vehicle, these often have more powerful transmitters, an external antenna and loudspeaker for handsfree use. They usually connect to the same networks as regular

mobile phones.

Cordless telephone (portable phone)
: Cordless phones are telephones which use one or more radio handsets in place of a wired handset. The handsets connect wirelessly to a base station, which in turn connects to a conventional land line for calling. Unlike mobile phones, cordless phones use private base stations (belonging to the land-line subscriber), and which are not shared.

Professional Mobile Radio
: Advanced professional mobile radio systems can be very similar to mobile phone systems. Notably, the IDEN standard has been used as both a private trunked radio system as well as the technology for several large public providers. Similar attempts have even been made to use TETRA, the European digital PMR standard, to implement public mobile networks.

Radio phone
: This is a term which covers radios which could connect into the telephone network. These phones may not be mobile; for example, they may require a mains power supply, they may require the assistance of a human operator to set up a PSTN phone call.

Satellite phone
: This type of phone communicates directly with an artificial satellite, which in turn relays calls to a base station or another satellite phone. A single satellite can provide coverage to a much greater area than terrestrial base stations. Since satellite phones are costly, their use is typically limited to people in remote areas where no mobile phone coverage exists, such as mountain climbers, mariners in the open sea, and news reporters at disaster sites.

WiFi Phones
: A relatively new type of mobile phone. These phones deliver calls over wireless internet networks as opposed to traditional CDMA and GSM network. Witel developed a WiFi phone in June 2009 that it retails for $59.99 making global calling affordable.

Usage

The cell phone novel is the first literary genre to emerge from the cellular age via text messaging to a website that collects the novels as a whole.[18] In virtual online computer games, readers can put themselves into first person in the story. Cell phone novels create a personal space for each individual reader. Paul Levinson, in Information on the Move (2004), says "...nowadays, a writer can write just about as easily, anywhere, as a reader can read" and they are "not only personal but portable".

Privacy

Cell phones have numerous privacy issues associated with them, and are regularly used by governments to perform surveillance.

Law enforcement and intelligence services in the U.K. and the United States possess technology to remotely activate the microphones in cell phones in order to listen to conversations that take place nearby the person who holds the phone.[19][20]

Mobile phones are also commonly used to collect location data. The geographical location of a mobile phone can be determined easily (whether it is being used or not), using a technique known multilateration to calculate the differences in time for a signal to travel from the cell phone to each of several cell towers near the owner of the phone. [21][22]

Health risks

Because mobile phones emit electromagnetic radiation, concerns have been raised about cancer risks that may pose when used for long periods of time.[23] This radiation is non-ionizing, but localized heating can occur.

The current consensus view of the scientific and medical communities is that health effects are very unlikely to be caused by cellular phones or their base stations.[24][25][26]

Cellular phones became widely available only relatively recently, while tumors can take decades to develop. For this reason, some health authorities have urged that the precautionary principle be observed, recommending that use and proximity to the head be minimized, especially by children.[27][28]

Controversial raw materials

Mobile phones and other electronic products have high quality capacitors in them, which contain tantalum. A major source of tantalum is the coltan ore from some illegal mines in the Democratic Republic of Congo operated by rebel groups to get money to fund their civil war.[29] A typical mobile phone has 40 milligrams of tantalum. A conflict-free source of tantalum are mines at Wodgina in the Pilbara region near Perth, Australia.[29]

Banning

Many places limit or restrict the use of mobile phones. Many schools has set restrictions on the use of mobile phones because of the use of cell phones for cheating on tests, harassing other people, causing threats to the schools security, and facilitating gossip and other social activity in school. Many mobile phones are banned in school locker room facilities and in public restrooms. New camera phones are required to have a shutter effect when a photo is taken.

Cell phone use is banned while driving in some states, either for adolescent drivers or everyone. Text messaging and wireless internet is banned while driving in most areas because of safety precautions.

See also

- Flexible keyboard
- Mobile telephony
- Mobile telecommunications
- Harvard sentences
- List of countries by number of mobile phones in use
- Mobile internet device (MID)
- Mobile Marketing Association
- ReCellular Inc.
- OpenBTS
- Information and communication technologies for development

References

1. ^ Ulyseas, Mark (2008-01-18). "Of Cigarettes and Cellphones (http://www.thebalitimes.com/2008/01/18/of-cigarettes-and-cellphones/)". The Bali Times. http://www.thebalitimes.com/2008/01/18/of-cigarettes-and-cellphones/. Retrieved on 2008-02-24.
2. ^ International Telecommunication Union. (2009). *Measuring the Information Society: The ICT Development Index (http://www.itu.int/ITU-D/ict/publications/idi/2009/material/IDI2009_w5.pdf)*. International Telecommunication

Union. pp. 108. ISBN 9261128319. http://www.itu.int/ITU-D/ict/publications/idi/2009/material/IDI2009_w5.pdf.

3. ^ Heeks, Richard (2008). "ICT4D 2.0: The Next Phase of Applying ICT for International Development (http://doi.ieeecomputersociety.org/10.1109/MC.2008.192)". *IEEE Computer* **41** (6): 26-33. http://doi.ieeecomputersociety.org/10.1109/MC.2008.192.
4. ^ "Special History Issue (http://wgbush.com/splncs/splncs15.pdf)" (PDF). *Speleonics 15* **IV** (3). October 1990. http://wgbush.com/splncs/splncs15.pdf.
5. ^ "Switching Plan for a Cellular Mobile Telephone System:, Z. Fluhr and E. Nussbaum, IEEE Transactions on Communications volume 21, #11 p. 1281 (1973)
6. ^ "Data signaling functions for a cellular mobile telephone system", V. Hachenburg, B. Holm and J. Smith, IEEE Trans Vehicular Technology, volume 26, #1 p. 82 (1977)
7. ^ Cooper, et al., "Radio Telephone System" (http://www.google.com/patents?id=nO8tAAAAEBAJ&dq=martin+cooper), US Patent number 3,906,166; Filing date: Oct 17, 1973; Issue date: September 1975; Assignee Motorola
8. ^ "Motorola Executive Helped spur Cellphone Revolution, Oversaw Ill-fated Iridium Project". *The Wall Street Journal*, June 20–21, 2009, p. A10.
9. ^ "John F. Mitchell, 1928-2009: Was president of Motorola from 1980 to '95, Chicago Tribune, June 17, 2009, retrieved June 17, 2009 (http://www.chicagotribune.com/news/chi-hed-jmitchell-17-jun17,0,955426.story)". Chicagotribune.com. http://www.chicagotribune.com/news/chi-hed-jmitchell-17-jun17,0,955426.story. Retrieved on 2009-07-29.
10. ^ Shiels, Maggie (2003-04-21). "BBC interview with Martin Cooper (http://news.bbc.co.uk/1/hi/uk/2963619.stm)". BBC News. http://news.bbc.co.uk/1/hi/uk/2963619.stm. Retrieved on 2009-07-29.
11. ^ "Swedish National Museum of Science and Technology (http://www.tekniskamuseet.se/mobilen/engelska/1980_90.shtml)". Tekniskamuseet.se. http://www.tekniskamuseet.se/mobilen/engelska/1980_90.shtml. Retrieved on 2009-07-29.
12. ^ UMTS World. "History of UMTS and 3G development (http://www.umtsworld.com/umts/history.htm)". Umtsworld.com. http://www.umtsworld.com/umts/history.htm. Retrieved on 2009-07-29.
13. ^ Reardon, Marguerite. "Nokia demos bendable cell phone" (http://news.cnet.com/8301-10784_3-9878005-7.html). CNET News, February 25, 2008. Retrieved 20 July 2009.
14. ^ "You Witness News (http://news.yahoo.com/you-witness-news)". News.yahoo.com. 2009-01-26. http://news.yahoo.com/you-witness-news. Retrieved on 2009-07-29.
15. ^ "GSM World agreement on Mobile phone Standard Charger (http://www.gsmworld.com/newsroom/press-releases/2009/2548.htm)". http://www.gsmworld.com/newsroom/press-releases/2009/2548.htm.
16. ^ "By the Numbers: Top Five Mobile Phone Vendors in the Third Quarter of 2008 - RCR Wireless News (http://www.rcrwireless.com/article/20081030/WIRELESS/810309997/)". Rcrwireless.com. 2009-07-21. http://www.rcrwireless.com/article/20081030/WIRELESS/810309997/. Retrieved on 2009-07-29.
17. ^ "Downloads_Guide (http://www.netsize.com/Ressources_Guide.htm)". Netsize. http://www.netsize.com/Ressources_Guide.htm. Retrieved on 2009-07-29.
18. ^ Goodyear, Dana (2009-01-07). "Letter from Japan: I ¢³⁄₄Novels (http://www.newyorker.com/reporting/2008/12/22/081222fa_fact_goodyear)". The New Yorker. http://www.newyorker.com/reporting/2008/12/22/081222fa_fact_goodyear. Retrieved on 2009-07-29.
19. ^ McCullagh, Declan; Anne Broache (December 1, 2006). "FBI taps cell phone mic as eavesdropping tool (http://news.cnet.com/FBI-taps-cell-phone-mic-as-eavesdropping-tool/2100-1029_3-6140191.html)". *CNet News*. http://news.cnet.com/FBI-taps-cell-phone-mic-as-eavesdropping-tool/2100-1029_3-6140191.html. Retrieved on 2009-03-14.
20. ^ Odell, Mark (August 1, 2005). "Use of mobile helped police keep tabs on suspect (http://news.ft.com/cms/s/7166b8a2-02cb-11da-84e5-00000e2511c8.html)". *Financial Times*. http://news.ft.com/cms/s/7166b8a2-02cb-11da-84e5-00000e2511c8.html. Retrieved on 2009-03-14.
21. ^ "Tracking a suspect by mobile phone (http://news.bbc.co.uk/1/hi/technology/4738219.stm)". *BBC News*. August 3, 2005. http://news.bbc.co.uk/1/hi/technology/4738219.stm. Retrieved on 2009-03-14.
22. ^ Miller, Joshua (March 14, 2009). "Cell Phone Tracking Can Locate Terrorists — But Only Where It's Legal (http://www.foxnews.com/story/0,2933,509211,00.html)". *FOX News*. http://www.foxnews.com/story/0,2933,509211,00.html. Retrieved on 2009-03-14.
23. ^ [Cellular Telephone Use and Cancer Risk http://www.cancer.gov/cancertopics/factsheet/risk/cellphones], National Cancer Institute.
24. ^ "What are the health risks associated with mobile phones and their base stations? (http://www.who.int/features/qa/30/en)". *Online Q&A*. World Health Organization. 2005-12-05. http://www.who.int/features/qa/30/en. Retrieved on 2008-01-19.
25. ^ "Electromagnetic fields and public health: mobile telephones and their base stations

(http://www.who.int/mediacentre/factsheets/fs193/en)". *Fact sheet N°193*. World Health Organization. June 2000. http://www.who.int/mediacentre/factsheets/fs193/en. Retrieved on 2008-01-19.

26. ^ Lönn, S; Ahlbom, A; Hall, P; Feychting, M (2005-03-15). "Long-Term Mobile Phone Use and Brain Tumor Risk (http://aje.oxfordjournals.org/cgi/content/full/161/6/526)". *American Journal of Epidemiology* (Oxford, UK: Oxford University Press) **161** (6): 526–535. doi:10.1093/aje/kwi091 (http://dx.doi.org/10.1093/aje/kwi091). doi:10.1093/aje/kwi091 (http://dx.doi.org/10.1093/aje/kwi091). ISSN 0002-9262 (http://worldcat.org/issn/0002-9262). OCLC 111065031 (http://worldcat.org/oclc/111065031). PMID 15746469 (http://www.ncbi.nlm.nih.gov/pubmed/15746469). http://aje.oxfordjournals.org/cgi/content/full/161/6/526. Retrieved on 2008-01-20.
27. ^ "Do mobile phones cause cancer? (http://www.cancerhelp.org.uk/help/default.asp?page=5265)". Cancer Research UK. 2008-12-05. http://www.cancerhelp.org.uk/help/default.asp?page=5265. Retrieved on 2009-05-27.
28. ^ Interlandi, Jeneen (2007-12-19). "How Safe Are Cell Phones? (http://www.newsweek.com/id/80966)". Newsweek. http://www.newsweek.com/id/80966. Retrieved on 2009-05-27.
29. ^ *a b* Hutcheon, Stephen. "Out of Africa: the blood tantalum in your mobile phone" (http://www.theage.com.au/news/home/technology/blood-tantalum-in-your-mobile/2009/05/08/1241289162634.html). *The Age*, May 8, 2009. Retrieved 20 July 2009.

Further reading

- Agar, Jon, *Constant Touch: A Global History of the Mobile Phone*, 2004 ISBN 1840465417
- Ahonen, Tomi, *m-Profits: Making Money with 3G Services*, 2002, ISBN 0-470-84775-1
- Ahonen, Kasper and Melkko, *3G Marketing* 2004, ISBN 0-470-85100-7
- Glotz, Peter & Bertsch, Stefan, eds. *Thumb Culture: The Meaning of Mobile Phones for Society*, 2005
- Katz, James E. & Aakhus, Mark, eds. *Perpetual Contact: Mobile Communication, Private Talk, Public Performance*, 2002
- Kavoori, Anandam & Arceneaux, Noah, eds. *The Cell Phone Reader: Essays in Social Transformation*, 2006
- Kopomaa, Timo. *The City in Your Pocket*, Gaudeamus 2000
- Levinson, Paul, *Cellphone: The Story of the World's Most Mobile Medium, and How It Has Transformed Everything!*, 2004 ISBN 1-4039-6041-0
- Ling, Rich, *The Mobile Connection: the Cell Phone's Impact on Society*, 2004 ISBN 1558609369
- Ling, Rich and Pedersen, Per, eds. *Mobile Communications: Re-negotiation of the Social Sphere*, 2005 ISBN 1852339314
- Home page of Rich Ling [1] (http://www.richardling.com/)
- Nyíri, Kristóf, ed. *Mobile Communication: Essays on Cognition and Community*, 2003
- Nyíri, Kristóf, ed. *Mobile Learning: Essays on Philosophy, Psychology and Education*, 2003
- Nyíri, Kristóf, ed. *Mobile Democracy: Essays on Society, Self and Politics*, 2003
- Nyíri, Kristóf, ed. *A Sense of Place: The Global and the Local in Mobile Communication*, 2005
- Nyíri, Kristóf, ed. *Mobile Understanding: The Epistemology of Ubiquitous Communication*, 2006
- Plant, Dr. Sadie, *on the mobile – the effects of mobile telephones on social and individual life* (http://www.motorola.com/mot/doc/0/234_MotDoc.pdf), 2001
- Rheingold, Howard, *Smart Mobs: The Next Social Revolution*, 2002 ISBN 0738208612
- Singh, Rohit (April 2009). *Mobile phones for development and profit: a win-win scenario (http://www.odi.org.uk/resources/odi-publications/opinions/128-mobile-phones-business-development-private-sector.pdf)*. Overseas Development Institute. pp. 2. http://www.odi.org.uk/resources/odi-publications/opinions/128-mobile-phones-business-development-private-sector.pdf.

External links

- How Cell Phones Work (http://www.howstuffworks.com/cell-phone.htm) at HowStuffWorks
- *Cell Phone, the ring heard around the world* (http://www.cbc.ca/doczone/cellphones/video.html)—a video documentary by the Canadian Broadcasting Corporation

Retrieved from "http://en.wikipedia.org/wiki/Mobile_phone"
Categories: Embedded systems | Mobile telecommunications

Hidden categories: Articles needing additional references from July 2009 | Articles needing additional references from February 2009 | All articles with unsourced statements | Articles with unsourced statements from March 2009 | All pages needing cleanup | Articles with specifically-marked weasel-worded phrases from July 2009 | Cleanup from section | Articles with unsourced statements from July 2009 | Wikipedia pages semi-protected against vandalism | Wikipedia protected pages without expiry

OIPE IAP40 JAN 04 2010 PATENT & TRADEMARK OFFICE

Best Available Copy

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
WASHINGTON, D.C. 20231
www.uspto.gov

APPLICATION NUMBER	FILING DATE	GRP ART UNIT	FIL FEE REC'D	ATTY DOCKET NO	DRAWINGS	TOT CLAIMS	IND CLAIMS
60/230,905	09/13/2000		75		4		

FILING RECEIPT

OC000000005589988

Donald Sidney Merrill
74 Chapel Hill Drive
Napa, CA 94559-2148

Date Mailed 12/01/2000

Receipt is acknowledged of this provisional Patent Application It will be considered in its order and you will be notified as to the results of the examination Be sure to provide the U.S APPLICATION NUMBER, FILING DATE, NAME OF APPLICANT, and TITLE OF INVENTION when inquiring about this application Fees transmitted by check or draft are subject to collection Please verify the accuracy of the data presented on this receipt **If an error is noted on this Filing Receipt, please write to the Office of Initial Patent Examination's Customer Service Center. Please provide a copy of this Filing Receipt with the changes noted thereon. If you received a "Notice to File Missing Parts" for this application, please submit any corrections to this Filing Receipt with your reply to the Notice. When the PTO processes the reply to the Notice, the PTO will generate another Filing Receipt incorporating the requested corrections (if appropriate).**

Applicant(s)

Donald Sidney Merrill, Napa, CA ,

Continuing Data as Claimed by Applicant

Foreign Applications

If Required, Foreign Filing License Granted 12/01/2000

**** SMALL ENTITY ****

Title

Electronic facilitation venue

Preliminary Class

Data entry by : THOMAS, DOROTHY **Team :** OIPE **Date:** 12/01/2000

LICENSE FOR FOREIGN FILING UNDER
Title 35, United States Code, Section 184
Title 37, Code of Federal Regulations, 5.11 & 5.15

GRANTED

The applicant has been granted a license under 35 U.S.C. 184, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" followed by a date appears on this form. Such licenses are issued in all applications where the conditions for issuance of a license have been met, regardless of whether or not a license may be required as set forth in 37 CRF 5 15 The scope and limitations of this license are set forth in 37 CFR 5 15(a) unless an earlier license has been issued under 37 CFR 5 15(b) The license is subject to revocation upon written notification The date indicated is the effective date of the license, unless an earlier license of similar scope has been granted under 37 CFR 5 13 or 5 14

This license is to be retained by the licensee and may be used at any time on or after the effective date thereof unless it is revoked This license is automatically transferred to any related applications(s) filed under 36 CFR 1 53(d) This license is not retroactive

The grant of a license does not in any way lessen the responsibility of a licensee for the security of the subject matter as imposed by any Government contract or the provisions of existing laws relating to espionage and the national security or the export of technical data Licensees should apprise themselves of current regulations especially with respect to certain countries, of other agencies, particularly the Office of Defense Trade Controls, Department of State (with respect to Arms, Munitions and Implements of War (22 CFR 121-128)), the Office of Export Administration, Department of Commerce (15 CFR 370 10 (j)); the Office of Foreign Assets Control, Department of Treasury (31 CFR Parts 500+) and the Department of Energy

NOT GRANTED

No license under 35 U S C 184 has been granted at this time, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" DOES NOT appear on this form Applicant may still petition for a license under 37 CFR 5 12, if a license is desired before the expiration of 6 months from the filing date of the application If 6 months has lapsed from the filing date of this application and the licensee has not received any indication of a secrecy order under 35 U S C. 181, the licensee may foreign file the application pursuant to 37 CFR 5 15(b)

PLEASE NOTE the following information about the Filing Receipt:

- The articles such as "**a**," "**an**" and "**the**" are not included as the first words in the title of an application. They are considered to be unnecessary to the understanding of the title
- The words "**new**," "**improved**," "**improvements in**" or "**relating to**" are not included as first words in the title of an application because a patent application, by nature, is a new idea or improvement
- The title may be truncated if it consists of more than 600 characters (letters and spaces combined)
- The docket number allows a maximum of 25 characters
- If your application was submitted under 37 CFR 1 10, your filing date should be the "date in" found on the Express Mail label. If there is a discrepancy, you should submit a request for a corrected Filing Receipt along with a copy of the Express Mail label showing the "date in "
- The title is recorded in sentence case

Any corrections that may need to be done to your Filing Receipt should be directed to

Assistant Commissioner for Patents
Office of Initial Patent Examination
Customer Service Center
Washington, DC 20231